# Système de sécurité biométrique multimodal par imagerie, dédié au contrôle d'accès

Pierre Bonazza

UBFC
UNIVERSITÉ
BOURGOGNE FRANCHE-COMTÉ

UNIVERSITÉ DE BOURGOGNE

**THÈSE DE DOCTORAT DE L'ÉTABLISSEMENT UNIVERSITÉ BOURGOGNE FRANCHE-COMTÉ**

**PRÉPARÉE À L'UNIVERSITÉ DE BOURGOGNE**

École doctorale n°37

Sciences Pour l'Ingénieur et Microtechniques

Doctorat d'Informatique et Instrumentation de l'Image

par

Pierre Bonazza

# Système de sécurité biométrique multimodal par imagerie, dédié au contrôle d'accès

Thèse présentée et soutenue à Dijon, le 21 Juin 2019

Composition du Jury :

| | | |
|---|---|---|
| Ducottet Christophe | Professeur à l'Université Jean Monnet de Saint Étienne | Rapporteur |
| Minghelli Audrey | Maître de Conférences HDR à l'Université de Toulon | Rapporteur |
| Madani Kurosh | Professeur à l'Université Paris-Est Créteil Val de Marne | Président du Jury |
| Dubois Julien | Maître de Conférences HDR à l'Université de Bourgogne | Directeur de thèse |
| Miteran Johel | Professeur à l'Université de Bourgogne | Codirecteur de thèse |
| Ginhac Dominique | Professeur à l'Université de Bourgogne | Codirecteur de thèse |

N° | X | X | X |

école doctorale **sciences pour l'ingénieur et microtechniques**

**Titre :** Système de sécurité biométrique multimodal par imagerie, dédié au contrôle d'accès

**Mots-clés :** Biométrie, Traitement d'image, Machine Learning, Deep Learning, Adéquation algorithmes/architecture

**Résumé :**

Ces travaux s'inscrivent dans le contexte d'un projet national visant à sécuriser le stockage et le transport de sources radioactives, lesquelles représentent un enjeu de sécurité important. L'objectif est de concevoir une solution technique répondant à la sécurisation de ces sources dans leur phase de stockage. La solution proposée, permettant une authentification des agents habilités, consiste en un système biométrique multimodal basé sur la vision et l'intelligence artificielle. Les recommandations actuelles relatives à l'utilisation de la biométrie en entreprise portent notamment sur la garantie de la vie privée des utilisateurs. La protection des données personnelles est donc une contrainte importante. Faisant appel au visage, aux empreintes digitales et aux vaisseaux sanguins du doigt, les modèles biométriques sont individuels et seulement stockés sur une carte sans contact nominative. Ce type de support ne disposant que d'une petite quantité de mémoire, l'axe de recherche privilégié repose sur une adéquation entre les algorithmes et l'architecture en charge des traitements. La quantité de données biométriques a ainsi été minimisée afin d'être stockées sur la carte sans contact. De nombreuses pistes ont été exploitées pour la modalité du visage, comparant alors des algorithmes issus du Machine Learning « classique » et d'autres issus du Deep Learning. Divers prétraitements ont été évalués afin de réduire l'influence de certaines variations environnementales sur l'acquisition. Les algorithmes traitant des deux modalités du doigt intègrent de nombreux prétraitements dont une banque de filtres de Gabor et une squelettisation. Ces prétraitements facilitent la détection de points d'intérêts propres à chaque modalité. Un descripteur décrit localement ces points et une mise en correspondance peut alors être effectuée entre des descripteurs de référence (stockés sur la carte sans contact) et des descripteurs issus de l'image acquise lors de l'authentification. Les attributs extraits de cette correspondance permettent, par une classification, de valider ou non l'authentification. Pour chacune des modalités étudiées, les données biométriques stockées sur la carte sans contact ne dépassent pas 2,6 Ko. Par la suite, une fusion de décision de ces trois modalités permet une authentification globale, inférant au système une meilleure robustesse aux potentielles intrusions ou usurpations d'identité. Avec un fort aspect industriel, ces travaux présentent également une intégration matérielle des méthodes développées. De plus, une sélection d'algorithmes est effectuée afin de permettre un fonctionnement temps réel de la phase d'authentification relativement à l'unité de calcul choisie.

**Title :** Système de sécurité biométrique multimodal par imagerie, dédié au contrôle d'accès

**Keywords:** Biometry, Image processing, Machine Learning, Deep Learning, algorithm/architecture matching

**Abstract:**

These thesis works are part of a national project aiming to secure storage and transports of radioactive sources, representing an important security issue. The objective is to design a technical solution addressing the secure of these sources in their storage phase. The proposed solution, allowing authorized staff person authentication, consists in a multimodal biometric security system, based on computer vision and artificial intelligence. The actual industrial biometric legal framework relates to preserve users' privacy. Therefore, the personal data protection is an important aspect. Using the face, fingerprint and finger veins modalities, the biometric models are individual and only stored on a nominative RFID card. This kind of remote media only have a small user's memory, thus, the leading research axis is focused on a matching between the algorithms and the processing unit in charge of the computation tasks. The amount of biometric data has then been minimized in order to be stored on the remote media. Manifold leads have been investigated for the face modality, with a comparison of classical Machine Learning algorithms and others from Deep Learning. Various preprocessing has been evaluated to lower the impact of environmental variations on the acquisition. The implemented algorithms dealing with both finger modalities include preprocessing algorithms, one of which are a bank of Gabor filters and a squeletonization. These preprocessing tasks help to detect points of interest. A descriptor locally describes these points and a matching is done then between references descriptors (stored on the remote card) and descriptors from the image acquired during the authentication. The attributes extracted from this matching provide, by a classification, a validation or a rejection of the authentication. For each studied modality, the biometric data stored on the remote media does not exceed 2,6 Ko. Thereafter, a decision fusion from these three modalities gives a global authentication which yields to a better robustness against intruders or spoofing attacks. With a strong industrial aspect, this work put forward a hardware integration of developed methods. Moreover, an algorithm selection ensures real time operation of the authentication task, with respect to the chosen computational unit.

UBFC UNIVERSITÉ BOURGOGNE FRANCHE-COMTÉ

Université Bourgogne Franche-Comté
Esplanade Erasme
21078, Dijon, France

> *L'important n'est pas de convaincre, mais de donner à réfléchir.*
>
> Bernard Werber, *Le père de nos pères*

# Remerciements

Ces travaux ont été financés par le Conseil Régional de Bourgogne Franche Comté (CBR). Le projet dans lequel s'inscrit cette thèse a été supporté par le pôle de l'industrie nucléaire, le Grand Chalon, BPI France, le conseil régional de l'Essonne et le pôle de compétitivité d'Île-de-France Systematic. Les principaux partenaires de ce projet ont été Assystem, Nuc21, ABC, Point-Core, Schlumberger, Systel Electronique, et l'Institut de Radioprotection et de Sûreté Nucléaire (IRSN).

Je tiens à remercier le laboratoire d'accueil ImViA (anciennement Le2i) et ses membres qui m'ont permis de réaliser cette thèse dans les meilleures conditions. Je remercie tout d'abord monsieur Julien Dubois, mon directeur de thèse, ainsi que messieurs Johel Miteran et Dominique Ginhac, co-directeurs de cette thèse, pour leurs précieux conseils, leur patience, leur bonne humeur en toutes situations et le temps qu'ils ont su m'accorder. Je remercie les enseignants chercheurs et autres permanents du laboratoire pour leur contribution non négligeable à l'atmosphère chaleureuse dans laquelle j'ai pu évoluer, en particulier madame Fan Yang Song et messieurs Yannick Benezeth et Barthélémy Heyrman. Je remercie également les membres du jury d'avoir accepté d'expertiser ces travaux : madame Audrey Minghelli et monsieur Christophe Ducottet, rapporteurs de cette thèse, ainsi que monsieur Kuroch Madani, examinateur lors de la soutenance.
Je remercie les membres de l'équipe technique, messieurs Cédric Clerc et Romain Troisgros pour le temps qu'ils ont su me consacrer, pour leur grande aide et leur précieux conseils sur l'élaboration du prototype et la conception des pièces. À monsieur Matthieu Rossé pour la conception des boitiers des Smart Camera Rasperry. À monsieur Vincent Thivent et sa société ODALID pour le prêt de son lecteur de cartes RFID. J'aimerai également exprimer ma gratitude envers monsieur David Darson, ingénieur de recherche au laboratoire de physique de l'École Normale Supérieure de Paris (LPENS), pour sa disponibilité, son expertise et ses conseils qui m'ont été d'une grande aide dans la conception du prototype de scanner rotatif présenté dans cette thèse.

Je souhaite remercier chaleureusement mes collègues et amis doctorants, pour avoir partagé ces trois années de thèse et sans qui elles n'auraient pas eu la même saveur. Je pense notamment à mesdames Yangjin Bi, Margarita Khokhlova et à messieurs Serge Bobbia, Yoan Marin, Anthony Boscaro, Axel Moinet, Nicolas Brochard, Richard Macwan, Roberto Marroquin, Romain Cendre.

Bien évidemment, je remercie mes parents pour m'avoir soutenu dans mes projets et mon parcours, et de m'avoir permis d'arriver jusque-là. Pour terminer, je tiens à exprimer toute ma reconnaissance à ma compagne, Audrey Febvre, pour avoir su me motiver et me supporter lors de cette thèse. Son soutien en toutes circonstances et ses encouragements m'ont accompagné sans interruption et ont sans aucun doute joué un rôle majeur dans l'aboutissement de cette thèse. Je dédie le fruit de ces travaux à mes parents, tout juste gradés grands-parents, à ma compagne qui est déjà une parfaite maman et à mon fils qui fait le bonheur de ses jeunes parents.

# Sommaire

# 1

# Introduction : motivations et sujet de recherche

Comme introduction à ces travaux, ce premier chapitre a pour objectif d'exposer le contexte lié à cette thèse et le projet national FUI dans lequel elle s'inscrit. De par la collaboration de différents acteurs, ce projet ayant pour nom Nuc-Track a pour objectif la sécurisation des phases transport et stockage de sources radioactives à usages civils. Utilisées dans des secteurs très variés, ces sources traduisent un enjeu de sureté nationale au travers de leurs différentes phases d'activité.

La contribution du laboratoire Le2i/ImViA au sein du projet Nuc-Track [1] concerne la phase stockage et consiste à développer un système d'authentification biométrique respectant un certain nombre de contraintes directement liées au projet et d'autres nécessaires au respect des normes en vigueur. Ce chapitre présente ce projet Nuc-Track ainsi que le cahier des charges en découlant, définissant les besoins et les contraintes de la problématique qui nous a été soumise. L'orientation de nos travaux de recherche et le cheminement logique de ce manuscrit sont également présentés.

## Sommaire

## 1.1/ Contexte

### 1.1.1/ Nuc-Track : Un projet national

En France, l'Institut de Radioprotection et de Sûreté Nucléaire (IRSN) et l'Autorité de Sûreté Nucléaire (ASN) sont chargés d'étudier les flux de transports de sources radioactives. Ils établissent les recommandations, ainsi que les restrictions s'imposant, du fait de l'enjeu de sûreté nationale que ces transports représentent. D'après leurs différentes fiches d'informations parues entre 2007 et 2017 [2, 3, 4], portant sur l'étude des flux de transports de substances radioactives à usage civil, environ 980000 sources radioactives circulent en France et près de 10 millions à l'international, générant approximativement 770000 mouvements par an dans le pays. Les transports sont répartis entre les voies terrestres, maritimes et aériennes. Représentant 96 % des mouvements [5], la route est un mode de transport privilégié, offrant plus de flexibilité, devant les transports ferroviaires dans la catégorie terrestre. Les mouvements de sources concernent divers secteurs d'activité tels que l'industrie/recherche nucléaire ou encore les domaines liés à la santé (table 1.1).

| **Secteur d'activité** | **Exemples de produits transportés** | **Exemples d'établissements concernés** | **Part des colis transportés** |
|---|---|---|---|
| Industrie Électronucléaire | Combustibles neufs ou usés, déchets nucléaires, outils contaminés | Installations du cycle du combustible, centres de recherches associées, centrales nucléaires | 16 % |
| Contrôles techniques et recherche | Appareils de détection de plomb, gammagraphes, sources non scellées utilisées comme traceurs radioactifs | Entreprises de diagnostic immobilier ou de gammagraphie, centres de recherches et universités | 56 % |
| Médecine | Produits radio-pharmaceutiques, sources de radiothérapie | Fournisseurs de sources, hôpitaux | 28 % |

Table 1.1 – Utilisation des sources radioactives à usages civil [3]

Ces sources ne représentent que 3 % du transport de matières dangereuses en France, ce qui n'en fait pas pour autant une opération moins risquée. Lors de ces mouvements on distingue en effet les trois phases "transport", "utilisation" et "stockage", et la responsabilité des sources est répartie entre de nombreux intermédiaires : l'expéditeur, le transporteur, le destinataire et le conseiller à la sécurité des transports.

Les principaux risques [2] liés à ces sources peuvent être résumés comme suit :

- **L'irradiation :** Les travailleurs ainsi que la population peuvent être exposés aux rayonnements. L'emballage doit offrir par conséquent une protection d'épaisseur adaptée à la nature et à l'intensité des rayonnements.
- **La contamination :** Les personnes peuvent être irradiées via le transfert de particules radioactives, que ce soit interne en cas d'ingestion ou d'inhalation, soit externe en cas de dépôt sur la peau ou sur le sol.
- **La criticité :** Des conditions particulières des matières fissiles peuvent amorcer une réaction en chaîne pouvant conduire à l'irradiation de personnes et au relâchement de radioéléments dans l'environnement.
- **Le vol ou le détournement :** Certaines matières sensibles peuvent être détournées à des fins malveillantes dont il est nécessaire de se prémunir.

Afin de prévenir au mieux la majorité de ces risques, les phases d'activité des sources sont soumises aux règlementations et recommandations de l'Agence Internationale de l'Énergie Atomique (AIEA) et de l'ASN, portées par les études de l'IRSN. Ces règles sont intégrées dans le "règlement type sur le transport des marchandises dangereuses", aussi appelé "livre orange", publié par l'Organisation des Nations Unies (ONU). Déclinées en plusieurs catégories suivant leur niveau de dangerosité, les sources sont stockées dans différents "emballages" adéquat, l'ensemble étant appelé "colis" (Figure 1.1). Bien qu'efficaces à l'encontre des premiers risques présentés ci-dessus, ces règles ne restreignent pas les possibilités de vol ou de détournement au strict minimum. Le gouvernement Français envisage donc de nouveaux arrêtés afin d'accroître la sécurité autour de ces sources, et de réduire les potentiels risques aussi bien internes qu'externes.

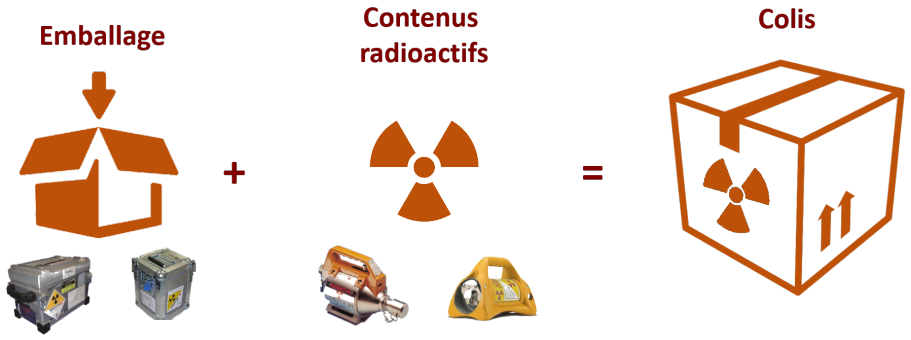



FIGURE 1.1 – Colis transporté

Dans le but de répondre à ces nouvelles directives et recommandations, différents acteurs français du domaine du nucléaire et de la sécurité collaborent sous la forme d'un projet national (Figure 1.2). Ainsi, le FUI "Nuc-Track" [1] a pour objectif de concevoir une solution technique aux recommandations et aux futures restrictions des autorités précédemment citées, répondant également aux contraintes des nombreux intermédiaires. En premier lieu, un cahier des charges précis a été établi définissant les différents besoins technologiques et juridiques. Un compromis a donc été nécessaire entre les spécifications techniques et les fonctionnalités envisagées, les besoins des utilisateurs et les nouvelles normes.


FIGURE 1.2 – Collaborateurs du projet Nuc-Track

### 1.1.2/ Solutions et contributions au projet Nuc-Track

La mission principale de Nuc-Track [6] est alors de développer un système complexe de suivi des sources en phase "transport" grâce aux technologies avancées dont nous disposons à l'heure actuelle (Figure 1.3). Ce système effectue en permanence divers contrôles tels que le suivi des transporteurs via une gestion complète des missions, la vérification du contenu des colis grâce à des capteurs ou encore une vérification GPS.
Dans une démarche d'anticipation, et pour se démarquer des autres dispositifs, nous nous sommes proposés de concevoir une solution répondant à la surveillance des sources en phase de stockage. Les zones de stockage pouvant être différentes suivant les secteurs d'activité, il est nécessaire de protéger leur accès ou de surveiller directement les sources "au repos" afin de réduire les potentielles menaces de vols ou de détérioration.

L'apport de la vision assistée par ordinateur nous apparaissant comme fondamental, nous avons considéré cette technologie pour explorer les différentes possibilités répondant à nos besoins. Une première solution se présente comme la surveillance directe des sources, ce qui implique de pouvoir les localiser et d'estimer la position des personnes dans l'espace *("pose estimation")* [7, 8] afin d'effectuer une reconnaissance d'actions. Une seconde, sur laquelle nous nous sommes focalisés, consiste à contrôler l'accès aux stockages via une vérification d'identité du personnel. Au vu du caractère sensible des produits surveillés, il est tout à fait envisageable lors de futurs travaux de faire cohabiter ces deux méthodes.

L'étude d'un système de sécurité biométrique nous a semblé réunir toutes les qualités nécessaires. En effet, ce type de contrôle d'accès présente l'avantage de vérifier précisément l'identité des personnes avant leur accès aux sources. Afin d'être implanté facilement dans n'importe quelle structure concernée, le système se doit d'avoir une mise en œuvre simpliste avec peu de ressources matérielles et humaines ainsi que des coûts de déploiement réduits. Le respect des différentes normes et recommandations, encadrant l'utilisation de la biométrie en environnement professionnel, est également un aspect à privilégier.

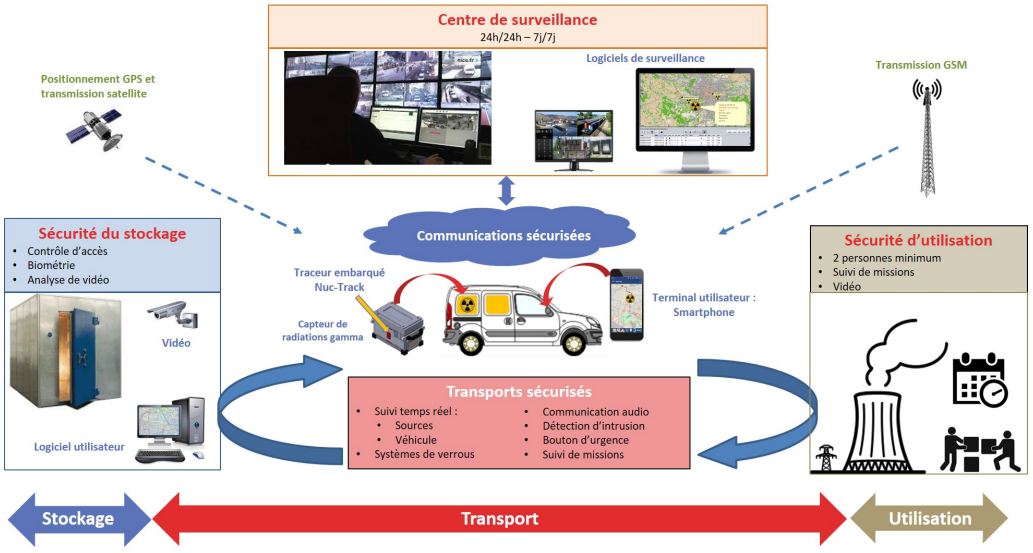



Figure 1.3 – Missions du Projet Nuc-Track

## 1.2/ Plan de recherche

### 1.2.1/ Conception d'un système de sécurité d'accès biométrique

Comme énoncé précédemment, l'objectif de cette thèse est de répondre à la problématique de protection des sources radioactives en phase de stockage, pour laquelle nous nous sommes orientés vers l'élaboration d'un système de sécurité biométrique. L'étude préliminaire sur la biométrie, présentée dans la section 2.1.1.1 du chapitre suivant, nous a amenés à considérer un système multimodal et d'ajouter en conséquence, aux exigences du cahier des charges, des contraintes issues de la législation encadrant la protection des données personnelles (§2.1.1.2 du chapitre 2). Vis-à-vis du projet, le système final doit imposer le moins de contraintes possible à l'usager *(collectabilité)*, tout en fournissant un niveau de sécurité suffisant au regard des sources *(performance et contournement)* et pouvoir s'interfacer aisément quel que soit l'emplacement du stockage. Afin de permettre une implantation simple sans maintenance particulière, le système doit comporter peu de capteurs, et procéder aux mesures via des algorithmes simples nécessitant le moins de puissance de calcul possible. D'un point de vue juridique, et par respect des normes et des recommandations de la CNIL, le système ne doit pas conserver de données personnelles ou les faire transiter sur le réseau. Les modalités devront également être acceptées des utilisateurs *(acceptabilité)*. Du fait de ces contraintes, nous avons choisi d'effectuer une "authentification" de personne, via l'utilisation d'un support personnel sans contact. L'utilisateur est donc le seul détenteur de ses données biométriques. Certaines données, nécessaires à la vérification d'identité mais sans lien direct avec la personne concernée, devront tout de même être stockées dans la mémoire du système, tel que le présente la figure 1.4. Les cartes sans contact, type RFID, ne possèdent qu'une faible capacité de stockage (quelques kilooctets), on ne peut donc pas négliger l'étude de la réduction de la taille des données biométriques, une problématique peu étudiée dans la littérature. Ces données, d'ores et déjà traitées afin d'en extraire les caractéristiques, puis miniaturisées (réduction de taille de l'espace de stockage), ne sont pas directement humainement interprétables et nécessitent la connaissance des algorithmes de traitement afin de remonter à l'information. La partie algorithmique est assurée directement par la caméra, embarquant une unité de calcul et le système de vision. Ainsi, la caméra devenue intelligente, et nommée en conséquence dans la littérature "Smart-Caméra", traite localement l'image acquise et ne transmet que le résultat de la mesure.

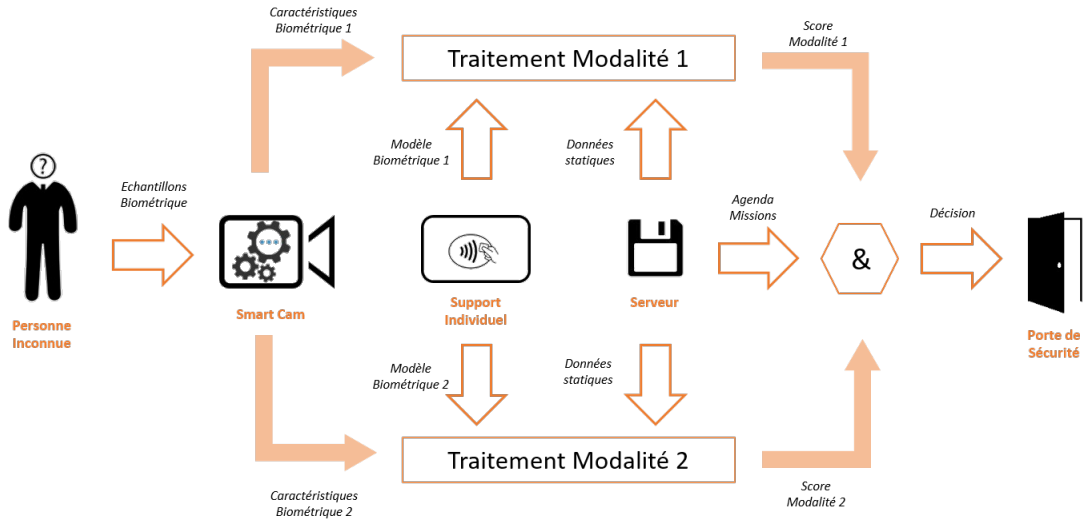



FIGURE 1.4 – Schéma de principe de notre système d'authentification biométrique

La première modalité choisie consiste en une mesure "standard" du visage de l'usager via l'utilisation d'une caméra 2D. L'étude consiste à trouver une certaine adéquation entre algorithmes et architecture, en comparant différents prétraitements, afin de minimiser l'impact des perturbations extérieures, et en confrontant différentes techniques de classification après avoir réduit la complexité des données. La deuxième modalité se trouve être une double mesure. En effet, acquises via un second système, ce sont deux nouvelles caractéristiques qui sont traitées. Celles-ci, situées toutes deux sur le doigt du sujet, correspondent au réseau vasculaire et aux empreintes digitales. L'étude ici, en considérant leur mesure sans contact, consiste à traiter chaque modalité indépendamment et éventuellement via une fusion de caractéristiques, ces deux attributs étant physiquement superposés sur le doigt. Pour l'utilisateur, ces deux dernières caractéristiques peuvent être considérées comme une seule, du fait de leur emplacement et de leur acquisition "transparente" (pas de changement de position de l'utilisateur) via le même capteur.

**Protocoles d'entraînement :**
Quelle que soit la modalité, il est nécessaire de passer par une phase d'enrôlement (ou entraînement du modèle biométrique). Ce qui consiste, comme illustré par la figure 1.5, à acquérir un échantillon et d'en extraire les caractéristiques puis d'entraîner un classifieur à les reconnaitre.

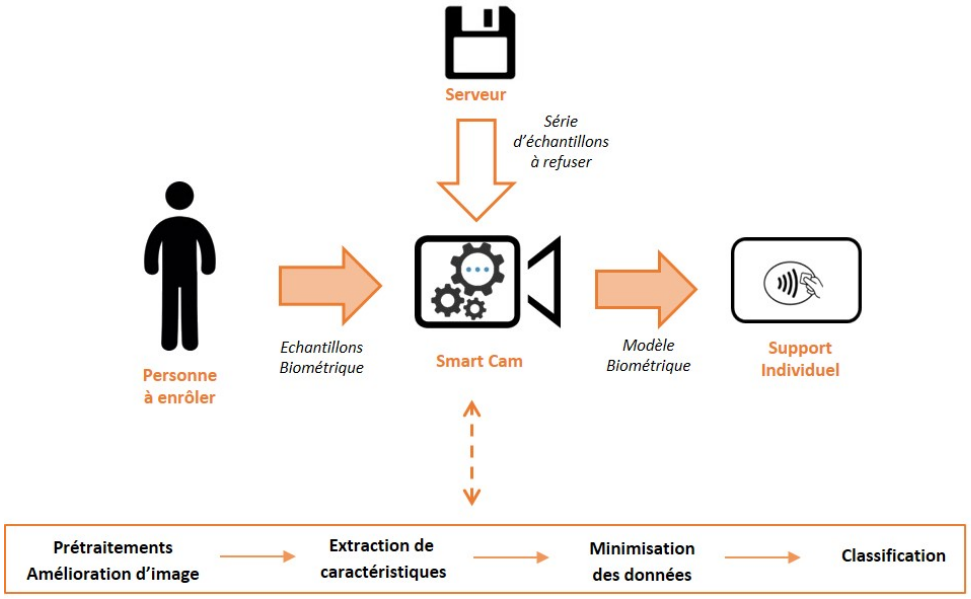



FIGURE 1.5 – Schéma du principe de l'enrôlement d'une nouvelle personne

Pour un système de reconnaissance biométrique, soit les caractéristiques extraites sont stockées en base de données afin d'être comparées à un nouvel échantillon lors du passage d'une personne inconnue, soit le modèle biométrique est construit à partir des caractéristiques de toutes les personnes à reconnaitre. Dans notre cas, qui est l'authentification, un modèle est créé par personne et est ensuite stocké sur le support sans contact de la personne en question. Le protocole d'entraînement utilisé dans cette thèse repose sur l'acquisition d'un certain nombre d'échantillons, par la caméra intelligente, de chacune des modalités.

Cette même caméra effectue les prétraitements nécessaires à la robustesse de la méthode quant aux diverses perturbations. Puis, en appliquant les mêmes traitements sur des échantillons "anonymes" représentant une moyenne de personnes à rejeter (provenant de bases de données de la littérature), une méthode de classification créée le modèle biométrique distinguant la personne à authentifier des personnes à refuser. Une attention particulière est portée à la minimisation des données dans le but d'obtenir un modèle léger en taille de stockage tout en conservant la fiabilité requise.

**Protocoles de tests :**
Pour authentifier une personne, il est nécessaire d'appliquer les mêmes prétraitements à ses échantillons que lors de la phase d'enrôlement. De la même manière que pour cette précédente phase, la caméra intelligente se charge d'acquérir et de "transformer" la modalité nécessaire à la vérification d'identité du porteur de la carte sans contact.

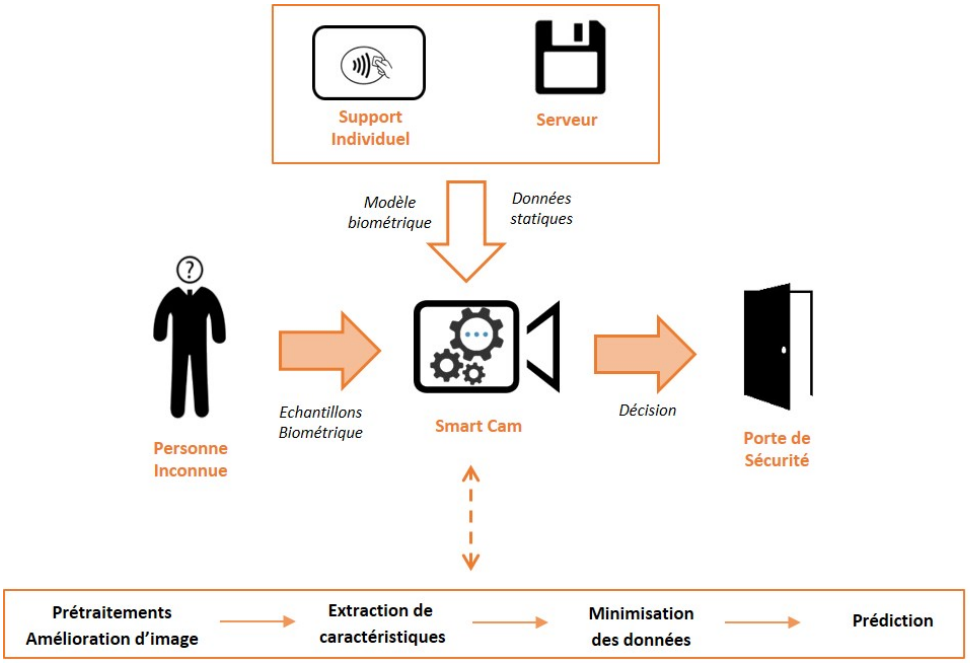



FIGURE 1.6 – Schéma du principe de l'authentification d'une personne

Un algorithme de "prédiction" propre au classifieur utilisé lors de l'enrôlement compare ensuite les nouveaux échantillons au modèle stocké sur la carte sans contact. Ces calculs donnent lieu à une prédiction, confirmant ou non l'identité, à laquelle est associée un score caractérisant la probabilité d'une réponse juste. Dans le cadre d'un système multi-biométrique (défini en 2.1.1.1), d'autres approches existent, consistant en une fusion à différents niveaux de la chaîne de traitement, en combinant par exemple les décisions finales, ou bien en appliquant une pondération aux scores. Ces méthodes de fusion sont présentées dans la section 2.2.1.2 .

### 1.2.2/ Construction du manuscrit

Dans ce manuscrit, nous présentons l'étude de différentes modalités biométriques dans un contexte de sécurisation d'accès à des zones de stockage de produits sensibles. Dans la continuité de cette introduction, le chapitre 2 présente la biométrie dans le domaine de la sécurité et illustre par des exemples les différentes modalités pouvant être mesurées. L'encadrement juridique propre à cette technologie est également détaillé, et plus particulièrement la législation dans le contexte particulièrement sensible de la protection des données personnelles. Afin de justifier nos choix en matière de biométrie, des systèmes biométriques multimodaux seront présentés et illustrés par des exemples du commerce pour les particuliers et les professionnels.
Dans le chapitre suivant (chapitre 3), un état de l'art général et non exhaustif de différentes techniques de Machine Learning destinées à la classification d'images est présenté. Ce chapitre traite du Machine Learning dit "classique" (technologie mature étudiée depuis de nombreuses années) et du Deep Learning (sous-ensemble plus récent du Machine Learning en constante progression) avec des Réseaux de Neurones Convolutifs (CNNs). Afin de faciliter la compréhension des résultats présentés dans les chapitres de nos contributions, différentes métriques d'évaluations utilisées dans la littérature sont exposées. Pour finir, nous définissons l'axe de recherche complet des études menées dans ce manuscrit en nous appuyant sur l'état de l'art présenté dans ces chapitres 2 et 3.

Les chapitres 4 et 5 présentent des applications plus spécifiques du Machine Learning, à savoir l'utilisation de ces algorithmes dans le cadre du projet lié à cette thèse. Le premier de ces deux chapitres traite des travaux engagés sur la modalité du visage. Nous comparons deux classifieurs classiques ayant fait leurs preuves (Machine à Vecteurs Support et Forêts Aléatoires) avec deux CNNs "légers" entraînés par "Transfer Learning". Le second chapitre porte sur l'étude de l'ensemble empreintes digitales et réseau vasculaire du doigt. Pour chacune de ces modalités, des points d'intérêts sont tout d'abord détectés et le descripteur BRIEF est appliqué. Puis, les ensembles de descriptions de deux images sont alors mis en correspondance et une classification est effectuée à partir des résultats de cette mesure. Dans ces deux chapitres, une attention particulière est portée à la minimisation de la quantité de données stockées, en vue de répondre aux réglementations et recommandations juridiques liées à la biométrie.

Un dernier chapitre clôture les contributions de cette thèse. Le chapitre 6 présente dans un premier temps une évaluation du système de sécurité biométrique complet, après sélection de la configuration optimale parmi les études des chapitres précédents. De par une fusion des trois modalités, une décision globale d'authentification est effectuée. Dans un second temps, le chapitre traite de l'implantation matérielle des développements effectués. Afin de vérifier la pertinence de la réduction de complexité des données biométriques, l'authentification biométrique du visage est implantée sur une caméra intelligente, composée d'une plateforme embarquée intégrant un processeur ARM. Les différents temps de traitement y sont mesurés afin de vérifier la contrainte temps réel. Ce chapitre présente également le développement d'un prototype original d'acquisition des empreintes digitales et du réseau vasculaire. Ce système se présente comme un scanner rotatif, mettant en rotation autour du doigt un ensemble d'éclairages proche infrarouge et une caméra linéaire infrarouge afin de mesurer rapidement ces deux modalités sur la périphérie du doigt.

Pour terminer, nous concluons par l'analyse critique des résultats, ouvrant la discussion sur les méthodes exploitées et les différentes possibilités existantes. Les nombreuses perspectives de ces études et les potentielles améliorations des travaux présentés sont également des aspects importants traités dans cette conclusion. Afin d'aider à la compréhension, des annexes permettent d'apporter de nombreuses précisions complémentaires sur les algorithmes utilisés.

# 2

# La biométrie dans le domaine de la sécurité

Afin d'établir un lien logique entre la problématique du projet et les choix effectués, précisant ainsi l'origine de certaines contraintes du cahier des charges, ce chapitre présente en premier lieu le concept de biométrie et les différentes caractéristiques humaines pouvant être mesurées. Les avantages et les inconvénients des modalités biométriques sont définis non seulement de par leur illustration au travers d'exemples, mais également par la présentation de l'encadrement juridique connexe à ce type de système.
La solution à la problématique du projet étant orientée vers un système de sécurité multimodal, ce chapitre présente également le principe de fusion des données biométriques à différents niveaux de la chaîne de traitement, ainsi que divers systèmes de sécurité biométriques multimodaux existants. Un choix est ensuite effectué quant aux modalités intégrées à notre système.

## Sommaire

## 2.1/ Systèmes de sécurité biométriques

### 2.1.1/ Généralités

#### 2.1.1.1/ Définitions et exemples

La biométrie consiste à définir l'identité d'une personne à l'aide de ses caractéristiques biométriques, que l'on appelle modalités, et que l'on classe parmi trois catégories : biologiques, morphologiques et comportementales. Inconsciemment, nous avons recours à la biométrie via nos capteurs sensoriels pour nous distinguer les uns des autres en reconnaissant le visage, la démarche, l'odeur ou encore la voix de nos proches. Les machines ne peuvent exploiter ces caractéristiques directement et, pour effectuer ces mesures à notre place, il est donc nécessaire de les détecter et de les formater de manière stable et distincte, à partir des échantillons fournis par le capteur utilisé. N'ayant pas de système permettant d'automatiser le processus, les premières études biométriques effectuées par l'Homme consistaient soit, pour les plus primitives, à mesurer les membres du sujet (circonférence de la tête, longueur des parties du corps, des pieds, du nez, etc.), soit à distinguer des points caractéristiques des empreintes digitales après les avoir posées sur un support avec de l'encre. Surtout nécessaires en criminologie, les empreintes digitales démontrent rapidement leur potentiel en matière de distinction des humains, du fait de leurs nombreuses particularités presque uniques pour chaque personne, et par leur faculté à "marquer" les objets touchés. En effet, il faut distinguer la biométrie dite "avec traces" de la biométrie dite "sans traces". Les empreintes digitales étant de la première catégorie il n'est pas toujours nécessaire d'avoir recours à des capteurs sophistiqués pour en effectuer l'acquisition.

Désormais, grâce aux avancées technologiques, un échantillon d'une modalité biométrique peut être comparé automatiquement et rapidement à plusieurs autres échantillons de cette même modalité, c'est ce que l'on appelle la "reconnaissance biométrique" (ou identification : 1 pour N). On la distingue de la "vérification biométrique" (ou authentification : 1 pour 1) qui consiste à comparer un échantillon à un autre et de confirmer ou non la correspondance. Dans le domaine de sécurité, la biométrie permet de protéger physiquement l'accès à un lieu, ou bien virtuellement l'accès à des données. Au travers d'une authentification ou une identification, cette technologie garantit un niveau de sécurité supérieur tout en s'affranchissant des risques d'oublis et de pertes liés à l'utilisation de mots de passe ou de clés physiques. Afin d'illustrer ces termes, nous pouvons prendre comme exemple le passeport biométrique, garant de l'identité d'une personne accédant à un pays grâce au visage et aux empreintes digitales de chaque doigt. Désormais implantée dans le secteur du grand public, la biométrie se retrouve également dans nos smartphones pour protéger l'accès aux données personnelles en utilisant les empreintes digitales comme protection, le visage ou même l'iris.

Les acquisitions sont majoritairement effectuées par le biais de la vision, mais d'autres types de capteurs peuvent être utilisés selon les besoins. De nos jours on répertorie de nombreuses modalités biométriques, les plus connues étant la mesure du visage, de l'iris, des empreintes digitales, de la voix, de l'ADN, etc. De nouvelles, plus originales et moins triviales, continuent d'émerger dans la littérature telles que la géométrie de l'oreille [9], le motif des lèvres [10] et des ongles [11], la thermographie [12] ou encore l'aspect du tracé du rythme cardiaque [13], etc. Les points suivants présentent, à titre d'illustration, un certain nombre des modalités existantes parmi les trois catégories nommées précédemment [14, 15], à savoir les caractéristiques morphologiques, comportementales et biologiques.

### Modalités Morphologiques :

— **Visage [16, 17] :**

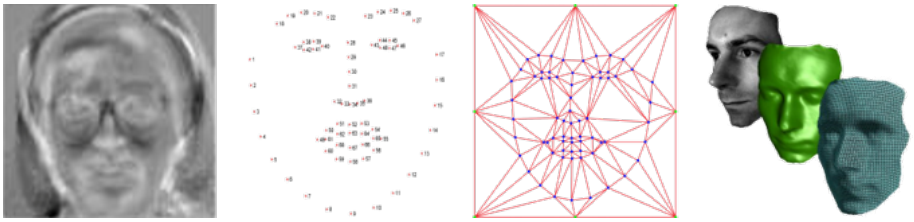


FIGURE 2.1 – Exemples de traitements pour la reconnaissance de visages [1,2,3]

La mesure de cette modalité est non intrusive (pas de contact physique) et bien acceptée du grand public du fait de sa similarité avec le processus humain de reconnaissance des personnes. Cette technique, considérée comme mature à l'heure actuelle, présente l'avantage de pouvoir utiliser les systèmes de vidéosurveillance existant si l'algorithme ne nécessite pas de mesures particulières. En effet, pour la plupart des traitements, une caméra standard suffit. Cependant, le système est alors soumis à certaines limitations telles que les variations de luminosité, les variations du visage (vieillissement, barbe, lunettes, expressions faciales, etc), l'angle de mesure ou contourné dans certains cas par l'usage d'un simple masque ou d'une photo. L'utilisation d'information 3D [18] (stéréovision, lumière structurée, etc) permet d'outrepasser la majorité de ces problèmes mais nécessite alors une implantation particulière et donc un coût supérieur de déploiement.
Afin de réduire la complexité des calculs et donc d'accélérer le traitement, une majorité des méthodes travaille avec des images en niveau de gris (1 canal). Les approches les plus populaires sont basées soit sur la localisation des points caractéristiques du visage (yeux, sourcils, nez, bouche, menton, etc.) et leur relation spatiale, soit sur l'analyse de l'aspect général d'une image représentant une combinaison linéaire d'un certain nombre de visages normés. D'autres informations complémentaires peuvent être estimées, comme le genre du sujet, son âge et la teinte de la peau, ou, en exploitant l'information thermique dans des longueurs d'onde plus lointaines.

— **Empreintes digitales [19] :**

FIGURE 2.2 – Exemples de reconnaissance d'empreintes digitales [4]

1. Image source : learnopencv.com/average-face
2. Image source : yimagesearch.com/facial-landmarks-dlib
3. Image source : cs.york.ac.uk/3Dface
4. Image source : identityone.net/BiometricTechnology

Très largement utilisée, cette modalité est une des plus anciennes utilisées et une des plus connues du grand public. La quantité d'informations disponibles est importante, mais seuls quelques points caractéristiques suffisent à discriminer une empreinte avec une grande fiabilité. La taille du modèle est donc réduite, nécessitant peu d'espace de stockage et permettant une reconnaissance rapide. Les empreintes digitales ont l'avantage d'être présentes sur chaque doigt de la main, ce qui en fait d'autant de motifs différents pour caractériser une personne. Les empreintes digitales ne changent pas de manière naturelle, ce qui garantit la pérennité des mesures dans le temps et réduit les risques de faux négatifs entre deux acquisitions. En revanche, des impuretés externes, des brûlures, des cicatrices, d'autres dommages ou des empreintes inexistantes peuvent être un obstacle à la reconnaissance. Les empreintes digitales peuvent également être usées ou altérées avec le temps selon le métier des sujets. La nécessité d'un contact physique avec le système d'acquisition pose problème sur la question de l'hygiène et peut-être un frein à l'usage pour certaines personnes.

— **Empreintes de la face palmaire de la main [20] :**

FIGURE 2.3 – Exemples d'empreinte de la face palmaire de la main [5]

Cette technique possède de nombreux points communs avec les empreintes digitales, tant au niveau de l'acquisition que de l'extraction des points caractéristiques (de nature similaire). Les motifs de surface d'un simple doigt fournissent déjà une grande quantité d'informations, mais cette information n'est qu'un échantillon partiel des données disponibles sur la main. La paume est constituée de diverses régions qui peuvent être scannées, stockées et traitées indépendamment selon les besoins : l'extrémité des doigts (empreintes digitales), la zone de jonction entre le creux de la main et les articulations des doigts, le creux de la main, la zone à la base du pouce ("éminence thénar"), la région contrôlant l'auriculaire ("éminence hypothénar") et le sillon distal du poignet.

— **Réseau vasculaire [21] :**
La mesure du réseau vasculaire s'effectue dans des longueurs d'ondes particulières du domaine proche infrarouge ([750 nm - 1000 nm]), pour lesquelles la lumière pénètre les tissus à faible profondeur pour atteindre les veines où le sang l'absorbe. Ces longueurs d'ondes, proches de la plage du visible, peuvent être captées par certains capteurs CMOS de qualité disposant d'un rendement quantique important, dépassant le domaine visible.

5. Image source : silicon.co.uk/security/biometric-payments

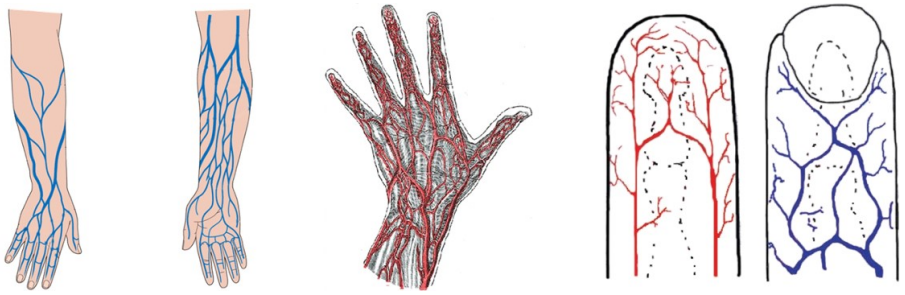


FIGURE 2.4 – Exemples de mesures de réseaux vasculaires [6, 7, 8]

Cette acquisition peut s'effectuer sur l'intérieur du bras, le dos de la main, ou l'intérieur d'un doigt. Le motif représenté par les veines est unique et permet de d'identifier une personne en détectant les bifurcations, les jonctions et la forme générale. Mais cette technique est encore peu connue du grand public. La mesure peut principalement être perturbée par la chaleur environnante et la température du corps.

— **Géométrie de la main [22] :**

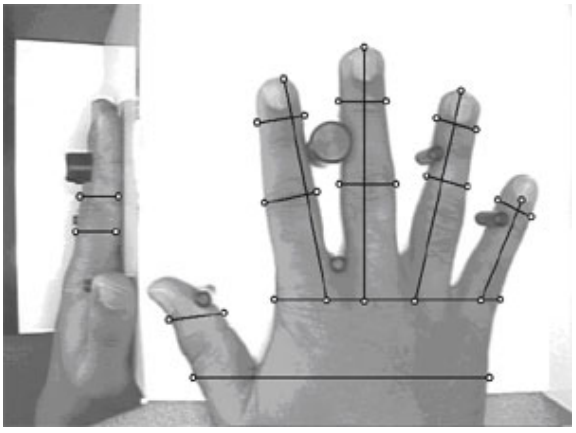


FIGURE 2.5 – Exemples d'acquisition de la géométrie de la main [9]

Contrairement à d'autres modalités, la mesure de la géométrie de la main n'est pas perturbée par la chaleur, l'humidité ou les impuretés de la peau et peut donc être effectuée dans n'importe quel environnement. L'acquisition ne nécessite pas d'effort particulier pour l'utilisateur, hormis souvent un positionnement particulier de la main autour de "marqueurs" dans le but de conserver la même position entre les mesures. Malgré tout, cette modalité est peu précise et présente un fort risque de similitude entre deux personnes.

— **Rétine [23] / Iris [24] :**
Ces deux modalités sont mesurées au niveau des yeux. L'iris est présent sur la face antérieure du globe oculaire et s'apparente à un "diaphragme" contrôlant l'intensité de lumière captée, tandis que la rétine se trouve sur la face interne et est l'organe sensible de la vision.

6. Image source : blackpoppymag.com/hands
7. Image source : cnet.com/Fujitsu-pay-with-your-veins
8. Image source : openi.nlm.nih.gov
9. Image source : optisec-systems.com/
10. Image source : m2sys.com/iris-recognition-vs-retina-scanning

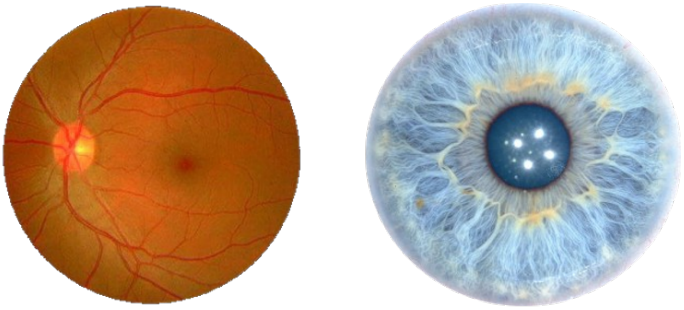


FIGURE 2.6 – Exemples de rétine et d'iris [10]

La mesure de la rétine s'effectue par l'acquisition du réseau vasculaire de la membrane épirétinienne. Cette technique de biométrie figure parmi les plus fiables en sécurité du fait de la stabilité de la rétine dans le temps, de la très forte unicité du motif rétinien et de la difficulté de falsification de ce dernier. En contrepartie, elle n'est que peu attrayante pour l'utilisateur de par les contraintes d'acquisition nécessitant un positionnement immobile et proche de la caméra. Les principales sources de perturbations intrinsèques au sujet sont liées à des maladies telles qu'un glaucome, du diabète ou encore une cataracte.
De la même manière que pour la rétine, un des avantages de la mesure de l'iris [24] est la grande unicité du motif. Hormis dans le cas de maladies telles que du diabète, l'iris demeure inchangée dans le temps. Elle présente l'avantage de posséder des caractéristiques extrêmement différentes d'une personne à l'autre. Le port de lunettes ou de lentilles de contact n'affecte pas la mesure tandis que la lumière ambiante amène des perturbations visibles. Les principales difficultés sont donc liées à l'éclairage, la distance du sujet au système et par conséquent la qualité de l'acquisition.

— **Oreille [25] :**

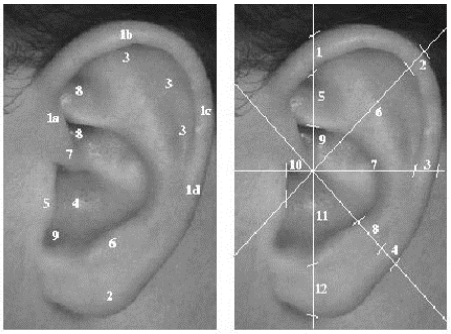



FIGURE 2.7 – Exemples de mesures pour la reconnaissance de l'oreille [11]

Contrairement à ce qu'on pourrait penser, car plutôt méconnue du grand public, les premiers travaux sur la mesure de l'oreille comme trait biométrique pour la reconnaissance de personne sont relativement anciens et datent de la fin du 19ème siècle. L'influence de facteurs aléatoires dans la constitution de la forme de l'oreille peut être observée en comparant les oreilles gauche et droite d'une même personne : bien que présentant des similarités, elles ne sont pas symétriques. Comme pour le visage, cette mesure s'effectue à distance mais peut facilement être perturbée par la présence des cheveux ou de boucles d'oreilles.

11. Image source : biometricsintegrated.com/ear-biometrics

**Modalités Comportementales :**

Contrairement aux modalités morphologiques qui ne nécessitent qu'une image ou une série d'images indépendantes pour détecter les caractéristiques nécessaires, les modalités comportementales impliquent l'utilisation de la vidéo ou d'une séquence d'images temporellement proches. Les exemples suivants n'ont pas vocation à être exhaustifs, et suffisent à illustrer cette catégorie. Il existe beaucoup d'autres modalités comportementales [26] telles que le mouvement des lèvres, le clignement des yeux, les mouvements d'utilisation d'une souris, le style de conduite ou encore le produit résultant d'une quelconque action (dessin, rédaction d'un texte, etc.).

— **Démarche [27] :**

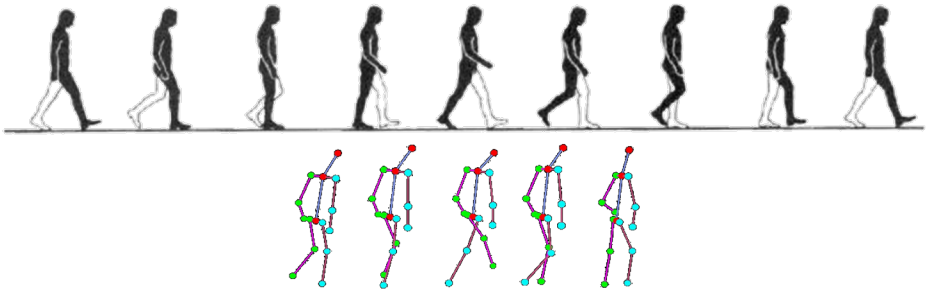


FIGURE 2.8 – Schémas des mesures de la démarche [12] [13]

Les systèmes de reconnaissance de la démarche nécessitent une séquence vidéo du sujet marchant, de profil ou de face, sur une certaine distance, issue d'une caméra 2D, d'une caméra 3D ou d'un ensemble de caméras sous différents angles. Sur chaque image, la silhouette humaine est détectée ainsi que les attributs spatiotemporels des extrémités et articulations du corps (tête, épaules, coudes, mains, hanches, genoux et pieds). Ainsi, par corrélation entre les mesures consécutives, cette modalité se base sur les estimations du balancier des bras, le rythme de la marche, le rebond, la longueur des pas, la distance entre la tête et les pieds ou encore la distance maximale entre les pieds droit et gauche. Les facteurs intrinsèques au sujet pouvant introduire un biais dans la mesure sont la prise de poids, une grossesse, l'état psychologique du sujet lors de la mesure, etc. Les chaussures, vêtements ou la surface de marche affectent également le procédé.

— **Signature [28] :**


FIGURE 2.9 – Exemple de mesures de la signature [14]

Il est question de mesurer la manière dont une personne signe à l'aide d'un stylet et d'une tablette graphique. En plus de l'aspect général de la signature, les caractéristiques mesurées incluent les pauses, les variations de rythme, la pression, la direction des traits et la vitesse. Cette signature change dans le temps et est influencée par la condition physique du sujet et le système d'écriture utilisé.

12. Image source : what-when-how.com/human-recognition-based-on-gait-poses-pattern
13. Image source : ee.oulu.fi/gait-recognition
14. Image source : tutorialspoint.com/behavioral-modalities

— **Dynamique de frappe au clavier [29] :**

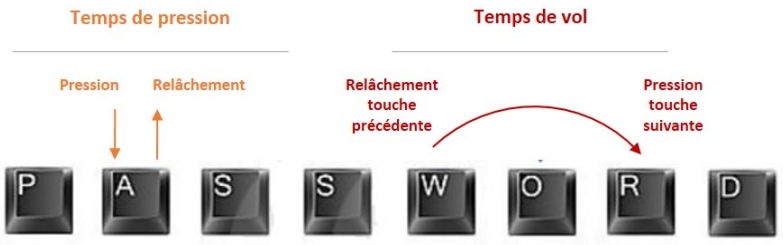



FIGURE 2.10 – Illustration des mesures de frappe au clavier

Les caractéristiques mesurées sont le temps de vol (temps entre deux frappes différentes), le temps de pression (temps entre une pression et le relâchement d'une même touche), la durée complète d'une séquence, la fréquence des erreurs, l'utilisation du pavé numérique, la force des frappes (pour les claviers équipés) et la façon d'utiliser les majuscules. Les mesures sont perturbées par l'état émotionnel du sujet, sa posture, le type de clavier, etc.

— **Voix [30] :**

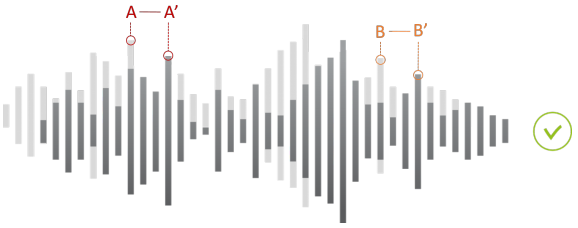



FIGURE 2.11 – Illustration de signaux vocaux [15]

Le son de notre voix est directement influencé par des facteurs physiques comme le nez, la bouche, les cordes vocales, ou par l'état émotionnel, la langue natale, les conditions médicales, et d'autres paramètres. La qualité de l'enregistrement dépend du capteur ou de l'environnement (écho, bruit, etc.). Il est possible d'isoler ces perturbations en utilisant plusieurs capteurs tout autour du sujet.

## Modalités Biologique :

— **ADN [31] :**

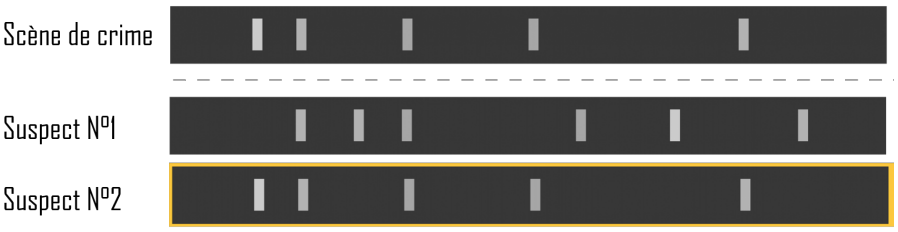



FIGURE 2.12 – Exemple d'empreintes ADN [16]

15. Image source : myforce.be/voice-biometry
16. Image source : yourgenome.org/what-is-dna-fingerprint

Présent dans toutes les cellules, l'Acide Désoxyribonucléique (ADN) est une macromolécule biologique contenant toute l'information génétique permettant le développement, le fonctionnement et la reproduction des êtres vivants. L'ADN extrait d'un quelconque échantillon biologique issu d'une personne (sang, salive, fragment de peau ou de poil, etc.) peut être utilisé pour déterminer son empreinte génétique. Le plus souvent utilisée en médecine légale, cette méthode isole et compare les séquences de segments d'ADN de différents individus, avec un risque de similitude entre deux personnes de moins d'un pour cent milliards. Très coûteuse en temps de traitement et en matériel, cette méthode n'est pas temps réel et peu commode dans le prélèvement des échantillons.

— **Électrocardiogramme [32] :**

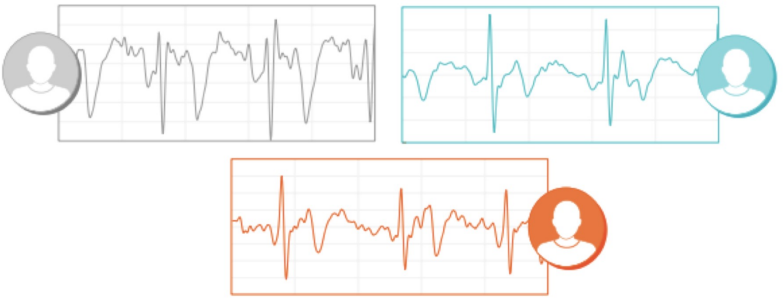


FIGURE 2.13 – Exemple d'électrocardiogrammes de 3 personnes [17]

Les mesures biométriques conventionnelles souffrent d'un risque de falsification via une reproduction de la modalité (masque pour le visage, empreintes digitales en silicone). Les mesures des signaux bioélectriques du corps comptent parmi des plus difficiles à contourner. L'électrocardiogramme (ECG) en fait partie, traduisant un changement du potentiel électrique des cellules cardiaques et possédant des caractéristiques uniques. Le motif de l'acquisition biométrique de l'ECG correspond à la dépolarisation et la repolarisation des battements de cœur. Les caractéristiques mesurées sur ces signaux représentent l'intervalle entre les extremums, leur amplitude, la pente et la largeur des pics.

On distingue sept particularités [33, 34] pour chaque attribut biométrique permettant de les comparer les unes aux autres et ainsi déterminer la plus appropriées à une situation donnée :

1. **L'universalité :** détermine si la modalité existe et si elle est présente quel que soit l'individu.
2. **L'unicité :** définit la probabilité de ne pas trouver de similitudes entre les mesures d'une même modalité sur des personnes différentes.
3. **La pérennité :** indique si la caractéristique reste inchangée sur une période donnée.
4. **La collectabilité :** détermine le degré de facilité de l'acquisition, de la mesure et de l'exploitation de la modalité,
5. **La performance :** caractérise la robustesse, la fiabilité et la vitesse de la mesure.
6. **L'acceptabilité :** indique si la modalité est acceptée sans objection par l'utilisateur.
7. **Le contournement :** représente la difficulté de contourner le système, par usurpation d'identité ou d'autres techniques de fraude.

17. Image source : sciencenewsforstudents.org/biometrics

| Modalité | Univ | Unic | Péren | Collec | Perf | Accept | Cont |
|---|---|---|---|---|---|---|---|
| **Visage** | *** | * | ** | *** | * | *** | * |
| **Empreintes digitales** | ** | *** | *** | ** | *** | ** | *** |
| **Paume de la main** | ** | *** | *** | ** | *** | ** | *** |
| **Réseau vasculaire** | ** | ** | ** | ** | ** | ** | *** |
| **Géométrie de la main** | ** | ** | ** | *** | ** | ** | ** |
| **Rétine** | *** | *** | ** | * | *** | * | *** |
| **Iris** | *** | *** | *** | ** | *** | * | *** |
| **Oreille** | ** | ** | *** | ** | ** | *** | ** |
| **Démarche** | ** | * | * | *** | * | *** | ** |
| **Signature** | * | * | * | *** | * | *** | * |
| **Frappe au clavier** | * | * | * | ** | * | ** | ** |
| **Voix** | ** | * | * | ** | * | *** | * |
| **ADN** | *** | *** | *** | * | *** | * | * |

TABLE 2.1 – Comparaison des modalités biométriques les plus connues [35] (*** représente les meilleures performances)

Les points précédents nous démontrent que certaines modalités, considérées seules, peuvent être peu performantes, que ce soit par une faible unicité, une falsification aisée ou des contraintes trop importantes. Une première approche permettant de réduire le nombre de faux positifs se base sur des capteurs complémentaires afin de contrôler les bonnes conditions d'acquisition (pas de falsification), comme par exemple en mesurant sans contact le rythme cardiaque [36, 37] sur le visage du sujet avant de l'authentifier. Une seconde approche pouvant améliorer les performances consiste en l'utilisation d'un système "multi-biométrique", pouvant prendre les formes suivantes (ou une combinaison de celles-ci) :

— **Multi-capteurs :** Une seule modalité est utilisée ici, mais mesurée par le biais de différents capteurs. Par exemple, pour acquérir la modalité sous différents angles simultanément, ou à différentes échelles, etc.
— **Multi-algorithmes :** La même mesure est traitée via différents algorithmes, permettant, via une simple acquisition, d'extraire des informations différentes. Cette approche est "économique" en coût matériel mais implique plus de complexité dans les calculs.
— **Multi-exemples :** Pour les modalités présentes à différents endroits du corps, plusieurs instances sont mesurées sur chacun d'eux. Comme par exemple l'acquisition de différentes empreintes digitales au lieu d'une, ou l'utilisation de l'iris ou de la rétine de chaque œil.
— **Multi-échantillons :** Plusieurs échantillons d'une même modalité sont mesurés par un seul capteur, en changeant la position autour du sujet, ou faisant varier l'éclairage par exemple.
— **Multi-modalités :** Différentes modalités sont combinées, via une fusion de décision sur chacune ou de par une fusion des caractéristiques extraites. Cette technique requiert des ressources algorithmiques et matérielles supplémentaires.

#### 2.1.1.2/ Encadrement juridique

Au-delà d'une question de sécurité, la biométrie présente un risque pour la vie privée des usagers. Cette technologie utilise des données caractéristiques propres à l'utilisateur et certaines, ne nécessitant pas de contact lors de l'acquisition, peuvent être utilisées à l'insu de ce dernier et ou à mauvais escient. Afin de limiter les potentielles dérives liées à la biométrie, la Commission Nationale de l'Informatique et des Libertés (CNIL), est chargée de faire valoir les droits et les devoirs des usagers dans le monde du numérique et de participer à l'élaboration d'un encadrement précis. Créée en 1978 par la loi Informatique et Libertés, la CNIL est une autorité administrative indépendante, constituée de 18 membres parmi lesquels on retrouve entre autres des parlementaires, le Président du Sénat, et des personnalités qualifiées désignées par le Président de l'Assemblée nationale. Elle dispose de 195 agents et de 16 millions d'euros de budget pour analyser, sur la vie privée et les libertés, l'impact des innovations technologiques et leurs usages émergents. Ses missions principales sont avant tout d'informer, de conseiller, d'accompagner et de protéger, mais également de contrôler et de sanctionner. Dans une démarche d'anticipation quant à la protection des données, elle travaille en étroite collaboration avec ses homologues européens et internationaux pour élaborer une régulation harmonisée.

Le 30 juin 2016, la CNIL a adopté deux autorisations uniques qui encadrent désormais l'ensemble des dispositifs de contrôle d'accès biométrique sur les lieux de travail, quels que soient les types de biométries utilisées. Elles distinguent les usages suivants :

- Les dispositifs biométriques permettant aux personnes de garder la maîtrise de leur gabarit biométrique *(AU-052* [18]*)*.
- Les dispositifs biométriques ne garantissant pas cette maîtrise *(AU-053* [19]*)*. Le principe de maîtrise du modèle biométrique est détaillé ci-après.

Au niveau Européen, la CNIL se réunit tous les deux mois à Bruxelles avec ses homologues européens au sein du G29 (Groupe de travail article 29 sur la protection des données). Le rôle de ce groupe de travail est de contribuer à l'élaboration des normes européennes, de rendre un avis sur le niveau de protection des pays hors Union Européenne et également de conseiller la Commission européenne. En 2018, le G29 se voit remplacé par un comité plus formel et cohérent : le Comité Européen de la Protection des Données (CEPD). Le 25 mai 2018, le règlement européen [20] sur la protection des données personnelles est entré en application [21], et de nombreuses formalités auprès de la CNIL disparaissent [22]. En contrepartie, la responsabilité des organismes usant de la biométrie est renforcée. Ils doivent désormais assurer une protection optimale des données à chaque instant et être en mesure de le démontrer en documentant leur conformité. La CNIL encadre de manière précise les demandes d'implantation de systèmes de sécurité biométrique, n'accordant une autorisation aux demandeurs qu'après vérification des obligations suivantes [23] :

- **Justifier que la biométrie soit pertinente :**
  Avant de mettre en place un contrôle d'accès, les organismes devront réfléchir à la pertinence du recours à un traitement biométrique. Ils devront documenter leur besoin, en expliquant notamment pourquoi un dispositif alternatif moins intrusif tel qu'un contrôle par badge, mot de passe, gardiennage ou vidéosurveillance ne suffit pas et pourquoi le dispositif biométrique répond à leurs besoins.

18. Legifrance AU-052 - NOR:CNIL1626033X
19. Legifrance AU-053 - NOR:CNIL1626008X
20. Site web : CNIL - Règlement du Parlement Européen
21. Legifrance, Protection des données personnelles, 2018 - NOR:JUSC1732261L
22. Site web : CNIL - Historique des modifications de loi
23. Site web : CNIL et biométrie - Principes à respecter

— **Privilégier les dispositifs permettant une maîtrise du gabarit :**
Les organismes devront privilégier les dispositifs garantissant par défaut, et dès leur conception, le contrôle de la personne concernée sur son gabarit, comme par exemple en conservant le gabarit biométrique sur un support individuel détenu par la seule personne concernée. L'objectif est de limiter le risque de détournement des données biométriques.
— **Justifier et documenter la conservation des gabarits en base :**
A défaut de pouvoir proposer les mécanismes protecteurs de la personne, comme explicité dans le point précédent, les organismes pourront faire appel à des systèmes reposant sur un stockage des gabarits en base après avoir rempli une grille d'analyse. La CNIL considère la centralisation des données biométriques dans des serveurs comme un risque majoré de détournement des données, aggravant les conséquences d'une potentielle défaillance du système. En cas d'attaque, c'est une multitude de données biométriques, et donc de personnes, qui pourraient être impactées. Le recours au stockage de gabarits biométriques en base n'est envisageable que s'il constitue la seule configuration répondant aux besoins spécifiques de l'organisme qui devra également adopter des mesures protégeant au maximum la vie privée, comme le chiffrement les données ou leur suppression en cas d'accès refusé.

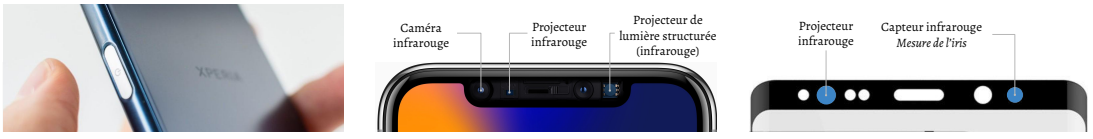
### 2.1.2/ Systèmes existant

Les systèmes de sécurité "classiques" présentent un défaut commun qui est de ne pas authentifier une personne mais plutôt un objet ou une connaissance qu'elle serait la seule à posséder (badge, code, etc). Les technologies biométriques apportent donc confort, simplicité ainsi qu'un haut niveau de sécurité aux utilisateurs, tout en présentant l'avantage de pouvoir être couplés aux systèmes "classiques" existants. En effet, que ce soit pour sécuriser un accès physique (un bâtiment, une pièce, un coffre-fort, etc) ou un accès logique (des données informatiques), la biométrie a déjà fait ses preuves et son utilisation est en constante progression. De nombreux systèmes se développent sur le marché, destinés au grand public ou au secteur privé (en entreprise avec une gestion horaire des accès), et proposent aussi bien des solutions unimodales que des solutions multimodales, que ce soit par le biais de différents terminaux d'acquisition ou par un unique terminal multicapteurs.

#### 2.1.2.1/ Systèmes grand public

La biométrie est déjà bien acceptée du grand public de par son intégration dans les *smartphones* [38, 39, 40] et les ordinateurs portables, pour sécuriser l'accès "physique" aux données personnelles, avec la démocratisation du déverrouillage par empreintes digitales, par le visage, par l'iris ou encore la voix. Les implémentations biométriques sur téléphones portables, illustrées par les exemples suivants, sont des méthodes hors ligne, dont les calculs sont strictement limités à la plateforme cible. Ces plateformes mettent en œuvre plusieurs dispositifs pour différentes modalités. Cependant, contrairement aux systèmes réellement multimodaux, une seule modalité est nécessaire pour déverrouiller le terminal. Étant essentiellement destinés à une seule personne et donc pour une authentification 1 pour 1, ces dispositifs peuvent enregistrer plusieurs "profils" et donc effectuer une identification 1 pour N. Ces dispositifs sont des sécurités additionnelles pour ces types appareils, dont la fonction première n'est pas la biométrie. Dans une optique de rapidité et de fiabilité supérieures, ils permettent de remplacer "temporairement" les traditionnels mots de passe, codes de déverrouillage ou les schémas, qui restent tout de même nécessaires en cas d'échec de la procédure biométrique. Ces systèmes biométriques doivent donc s'interfacer au *smartphone* et faire face à certaines contraintes : puissance nécessaire et taille d'intégration si l'on considère l'utilisation des capteurs supplémentaires.

Désormais, la quasi-totalité des *smartphones* embarquent un capteur pour la reconnaissance d'empreintes. La figure 2.14a donne en exemple le bouton latéral marche/arrêt (4 mm x 9 mm) d'un téléphone *Sony* auquel est associé un capteur d'empreinte, permettant un déverrouillage instantané du terminal. La recherche actuelle pour cette modalité embarquée sur *smartphone* porte sur l'intégration d'un capteur d'empreintes couvrant la totalité de l'écran tactile [41, 42, 43, 44], le rendant alors capable de mesurer cette modalité de manière totalement transparente en continu.
De même, la reconnaissance de visage peut être utilisée par n'importe quel *smartphone* du fait qu'elle ne nécessite aucun capteur particulier, hormis une caméra dont tous les terminaux sont équipés aujourd'hui (à l'avant et à l'arrière). En revanche, la société *Apple* a proposé en 2017 une avancée conséquente pour cette modalité, permettant d'effectuer une reconnaissance du visage en trois dimensions avec un simple *smartphone* à une distance de 20 cm. Nommée *FaceID* ou *True Depth* [45]) et illustrée par la figure 2.14b, cette technologie intègre, sur une surface très restreinte (3,5 cm x 1,2 cm), un projecteur infrarouge à lumière structurée permettant de cartographier le visage au travers d'une matrice de 30000 points et une caméra infrarouge dont la tâche est de relever la déformation de ces points due aux traits du visage. Il est ainsi impossible de tromper le système avec une simple photo, et le processus fonctionne également dans n'importe quelles conditions d'éclairage du fait de ses capacités infrarouges.
La même année, l'entreprise *Samsung* propose d'intégrer un scanner d'iris sur ces *smartphones*. Illustré figure 2.14c, le terminal en question est également muni d'une caméra infrarouge mais d'un simple projecteur LED infrarouge, lui permettant d'acquérir une image infrarouge de la région de l'œil du sujet (également sous n'importe quelle condition d'éclairage) et d'effectuer les traitements nécessaires à la détection et la reconnaissance de l'iris [46]. Contrairement au système précédent, celui-ci est plus contraignant pour l'utilisateur, nécessitant un bon positionnement de l'œil envers la caméra, et peut être trompé avec une photo de bonne qualité.




(a) Sony[24] - Empreintes (b) Apple[25] - FaceID (c) Samsung[26] - Scanner d'Iris

FIGURE 2.14 – Exemples de dispositifs biométriques intégrés aux *smartphones*

Il y a donc un vaste panel de dispositifs biométriques dédiés au grand public embarqués dans nos *smartphones*. Les terminaux portables intégrant plusieurs solutions pour différentes modalités, le choix revient à l'utilisateur de considérer la méthode qui lui semble la plus adéquate. En effet, pour le grand public, avec une moyenne de 58 déverrouillages du téléphone par jour en 2014 [47], un compromis doit être fait entre contraintes d'utilisation, sécurité et vitesse d'exécution. De ce fait, c'est cette dernière condition qui est la plus considérée dans ce choix pour un tel usage journalier. Ainsi, avec un déverrouillage instantané, ce sont les empreintes digitales qui demeurent une solution préférable aux autres modalités [48]. En comparaison, l'exemple précédent du *FaceID* est plus sécurisé, mais nécessite une contrainte supplémentaire quant à la position du sujet par rapport à la caméra, et le temps de déverrouillage peut atteindre 1,5 secondes. L'intégration de capteurs particuliers augmente également les coûts de productions et de maintenance.

24. Sony Xperia XZ1 : Bouton latéral ON/OFF et capteur d'empreinte
25. Apple iPhone X : Capteurs avant du FaceID *TrueDepth*
26. Samsung Galaxy S9 : Capteur avant pour la reconnaissance de visage et scanner d'iris

Les systèmes biométriques pour le grand public ne sont pas limités aux cas très répandu des *smartphones*, et des systèmes pour la maison leurs sont également proposés. Nous pouvons prendre pour exemple deux dispositifs de l'entreprise française Netatmo présentés figure 2.15. Conçus en tant qu'objets connectés pour la surveillance de la maison, ils disposent de fonctions de communications avec une gestion d'évènements afin d'avertir les propriétaires sur leur *smartphone*. La caméra destinée à un usage intérieur, figure 2.15a, effectue une reconnaissance de visages et notifie de la présence des membres de la famille ou de personnes inconnues. La caméra à usage extérieur, figure 2.15b, n'effectue pas de reconnaissance biométrique mais est capable de capter la présence d'une voiture, de personnes ou d'animaux.

(a) "Welcome[27]" - Caméra intérieure

(b) "Presence[28]" - Caméra extérieure

FIGURE 2.15 – Caméras intelligentes intérieure et extérieure Netatmo

#### 2.1.2.2/ SYSTÈMES INDUSTRIELS

Dans le cas des systèmes destinés aux installations professionnelles, le compromis précédemment cité est toujours applicable, néanmoins c'est la sécurité qui est mise en avant, plus que les contraintes d'utilisation et la vitesse qui sont tout de même des facteurs pris en compte. Ici, ces systèmes sont fixes et sont le plus souvent utilisés pour sécuriser l'accès à une zone restreinte. Ils peuvent considérer une seule modalité ou plusieurs (multimodal) et prendre les différentes formes citées dans la section 2.1.1.1, ou une combinaison de ces formes, à savoir multicapteurs, multi-algorithmes, multi-exemples, multi-échantillons et multimodalité. Contrairement aux *smartphones*, les systèmes multimodaux nécessitent la mesure de chacune d'elles pour valider l'accès, l'utilisateur n'a donc plus de choix à envisager quant à la mesure à effectuer.
Ces systèmes dédiés aux professionnels peuvent intégrer les données biométriques directement dans la mémoire du système, ou bien elles peuvent être stockées sur un serveur afin qu'un même modèle puisse alimenter plusieurs dispositifs (un bâtiment utilisant un même système pour différentes zones). Ainsi, une identification peut être effectuée au même titre qu'une authentification. Dans la majeure partie des cas où ces systèmes nécessitent une carte personnelle sans contact, celle-ci transmet simplement l'identifiant de son propriétaire qui est alors utilisé pour gérer un agenda d'accès et/ou effectuer une authentification en ne comparant la mesure biométrique qu'au modèle correspondant à cet identifiant dans la base de données. Certaines modalités, comme les empreintes digitales, fournissent des caractéristiques biométriques très légères, qui peuvent alors être stockées sur une carte sans contact, laquelle ne bénéficiant que d'une très faible capacité de stockage. Ces dispositifs étant entièrement dédiés à la sécurisation d'accès, leur conception est orientée pour une intégration optimale de la modalité utilisée. Ainsi, contrairement aux *smartphones*, la forme, la taille et la partie calculatoire peuvent être adaptées en fonction des besoins.

27. Netatmo Welcome - Caméra intelligente d'intérieur avec reconnaissance faciale
28. Netatmo Presence - Caméra intelligente d'extérieur avec reconnaissance d'objets

La figure 2.16 présente en exemple quelques systèmes de sécurité d'accès biométrique, destinés aux professionnels, parmi les modalités les plus utilisées à savoir le visage et les empreintes digitales, ainsi que la fusion des empreintes digitales et du réseau vasculaire du doigt.

**Visage**

(a) FaceStation 2[29] [49]

(d) 3D Face Reader[31] [52]

**Empreintes digitales**

(b) BioStation 2[29] [50]

(e) Wave Tower[31] [53]

**Systèmes hybrides (Empreintes et veines)**

(c) Hybrid Finger Scanner[29] [51]


(f) Smart Finger VP[31] [54]

FIGURE 2.16 – Exemples de systèmes biométriques pour les professionnels

Les systèmes "FaceStation 2" [49] et "BioStation 2" [50], figures 2.16a et 2.16b, font partie des solutions biométriques Suprema, supportant l'authentification ainsi que la reconnaissance de personnes. Ces deux systèmes disposent d'une interface utilisateur permettant de communiquer des instructions et de signaler un dysfonctionnement, ainsi que d'un lecteur intégré de cartes sans contact. Cette carte permet d'obtenir l'identifiant de son possesseur et de procéder par la suite à une authentification, les modèles biométriques des usagers étant stockés au sein du système (jusqu'à 30000 utilisateurs pour le visage et 1 million pour les empreintes digitales) ou sur un serveur. Les systèmes "3D face reader" [52] et "Wave Tower" [53], figures 2.16d et 2.16e, font partie des solutions biométrique IDEMIA. De la même manière que le FaceID, le système "3D face reader" permet une acquisition en 3 dimensions du visage grâce la projection de 40000 points via un projecteur à lumière infrarouge structurée. Ce système peut contenir jusqu'à trois mille profils en mode identification (reconnaissance) et cent mille profils en mode authentification via l'usage d'une carte sans contact transmettant un identifiant. Le système "Wave Tower" relève les empreintes digitales du sujet. Sans contact, le système est capable d'acquérir les empreintes digitales de quatre doigts de la main en les faisant simplement survoler le lecteur. Les caractéristiques des empreintes digitales étant plus "légères" que celles du visage, ce système peut contenir quarante mille profils en mode identification et un million en mode authentification toujours via l'usage d'un badge sans contact.

29. Solutions biométriques Suprema
30. Solutions biométriques NEC
31. Solutions biométriques Safran-Morpho (IDEMIA)

Les deux systèmes suivants sont des cas particuliers, permettant la mesure simultanée de deux modalités avec un seul système d'acquisition disposant de deux capteurs. Ces deux systèmes, "Hybrid Finger Scanner" de NEC [51] et "Smart Finger VP" de IDEMIA [54], permettent d'acquérir les empreintes digitales (se trouvant sur la première phalange d'un doigt) ainsi que le réseau vasculaire à hauteur de la deuxième et troisième phalange. Ainsi, ces dispositifs peuvent effectuer un contrôle sur chacune de ces modalités mais également sur les caractéristiques d'un nouveau motif issu de la fusion des caractéristiques précédentes. L'apport de l'information sous-cutanée, plus difficile à mesurer et donc plus difficile à copier, permet un renfort de la sécurité.
Bien sûr, les exemples illustrés ici ne sont pas exhaustifs. D'autres systèmes sont capables de mesurer le réseau vasculaire de la paume de la main, tels que les solutions Zalix avec "VeinAccess Intellegent Security" [55], ou encore la gamme de m2sys [56], proposant différents systèmes d'acquisition (veines du doigt, empreintes digitales, iris, veines de dos de la main) pouvant s'interfacer ensemble et ainsi fournir des résultats complémentaires à chacune de ces modalités indépendantes.

## 2.2/ Approche multimodale

### 2.2.1/ Multimodalité

#### 2.2.1.1/ Principe

Les systèmes de sécurité biométrique à modalité unique, bien qu'ayant été considérablement améliorés ces dernières années de par des algorithmes plus performants et des capteurs plus précis, souffrent de certains problèmes de fiabilité notamment liés aux critères de contournement et d'unicité, souvent trop faibles selon les modalités. Par conséquent, les systèmes biométriques multimodaux sont au cœur des solutions les plus performantes en matière de sécurité, permettant de combiner plusieurs modalités (le plus souvent deux) acquises simultanément avec un ou plusieurs dispositifs de mesure [57]. Le système est ainsi capable d'effectuer un contrôle de sécurité sur un nouveau jeu de caractéristiques émanant de la fusion des attributs de ces modalités. Il est également possible de compléter ce contrôle en traitant indépendamment en parallèle les caractéristiques de chacune des modalités mesurées. Un système multimodal est donc en mesure de compenser les lacunes des systèmes uni-modaux, en proposant une solution plus difficilement falsifiable et plus robuste aux similitudes entre les individus ainsi qu'aux données d'acquisition bruitées. En effet, il est indéniable qu'un usurpateur aura plus de mal à copier plusieurs modalités d'une même personne. Il est également peu probable que plusieurs modalités soient similaires entre deux personnes. Cette fusion peut s'effectuer à différents niveaux des chaînes de traitement et de décision.

#### 2.2.1.2/ Fusion de données biométriques

Comme présenté dans la section 2.1.1.1, la biométrie n'est pas restreinte à la mesure d'une modalité par un capteur, mais peut résulter de la fusion d'informations provenant de sources variées, comme le montre la figure 2.17, sous les formes suivantes d'acquisition : multi-capteurs, multi-algorithmes, multi-exemples, multi-échantillons. Cette liste est valable que ce soit pour des systèmes uni-modaux ou systèmes multimodaux [58]. Chacune des modalités traitées par les systèmes multimodaux peut être vue comme une unique modalité et les informations utilisées peuvent provenir des sources citées précédemment. L'étude menée par Suneet Narula Garg et.al [59] confirme l'apport non négligeable de la fusion multimodale face à une unique modalité, augmentant significativement les performances d'authentification en comparaison des performances de chacune des modalités mesurées indépendamment les unes des autres [60].

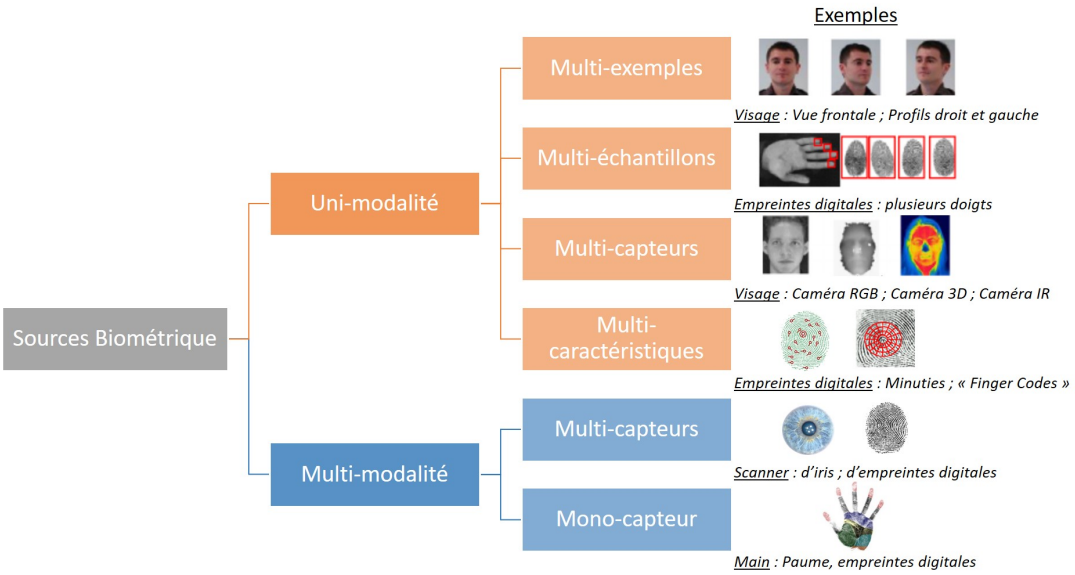



FIGURE 2.17 – Sources d'informations dans un système de fusion biométrique [58]

La fusion biométrique intervient dans la combinaison d'informations biométriques d'une ou plusieurs modalités afin de renforcer les atouts et réduire les lacunes de ces modalités, induits de ces informations. On distingue différents types de fusions biométriques, répartis à différents niveaux de la chaîne du système [61] (comprenant l'acquisition de l'information, son traitement, l'extraction de caractéristiques puis la classification de laquelle découle une décision), que l'on peut séparer en deux grandes familles : avant la classification et après la classification [62, 59]. Les différents niveaux de fusion sont résumés par la figure 2.18.

### Avant classification

Avant la classification (*matching*), la combinaison des informations peut avoir lieu soit au niveau capteur, soit au niveau des caractéristiques.

- **Fusion au niveau capteur *(Sensor level)* :**
  La fusion au niveau capteur [63] s'effectue à partir des données brutes issues d'un ou plusieurs capteurs. Les données doivent par conséquent être compatibles entre elles (de même type) et donc provenir de capteurs similaires si plusieurs sont utilisés. On peut illustrer cette fusion par l'acquisition du signal de la voix par deux capteurs (microphones) ou plus, situés à différents endroits dans l'espace. Dans le cas de l'image, il est nécessaire que les informations à fusionner soient de la même dimension.
- **Fusion au niveau caractéristiques *(Feature level)* :**
  À ce niveau de la chaîne de traitement, les différentes instances d'une même modalité biométrique, ou les différentes modalités, ont été acquises par un ou plusieurs capteurs. De l'extraction de caractéristiques résulte, pour chacune des mesures, un vecteur décrivant ces caractéristiques et la fusion au niveau caractéristiques consiste à combiner ces vecteurs. Des vecteurs de caractéristiques homogènes (plusieurs mesures d'une même modalité) permettent de construire un unique vecteur à partir de leur somme pondérée. Des vecteurs hétérogènes (mesures de différentes modalités) permettent également de construire un unique vecteur mais en concaténant ces vecteurs individuels. Ainsi, le vecteur résultant est très riche en informations. Cependant, ce type de fusion peut entraîner la création d'un vecteur de très grande dimension et nécessiter une puissance de calcul importante.

### Après classification

Après la classification, l'intégration de multiples informations biométriques peut intervenir au travers de deux catégories : la sélection dynamique de classifieurs et la fusion de classifieurs [64].

- **Sélection dynamique de classifieurs *(Dynamic classifier selection)* :**
  La sélection dynamique de classifieurs, aussi connue sous le nom de l'approche *"winner-take-all"*, nécessite de considérer un partitionnement des données d'entrée [65]. Ces partitions peuvent être définies par l'ensemble des décisions individuelles de chaque classifieur, ou par les caractéristiques des échantillons d'entrée. Pour chaque partition, le classifieur fournissant les résultats optimaux sur des données d'apprentissage ou de validation est sélectionné. Pour la classification, un échantillon inconnu est affecté à une partition et c'est la décision du classifieur, associé à cette partition, le plus à même de donner une décision correcte, qui est utilisée pour la décision finale. Pour résumer, la sélection dynamique de classificateurs tente de prédire quel classificateur est le plus susceptible de fournir un résultat correct pour un échantillon donné.
- **Fusion de classifieurs *(Classifier fusion)* :**
  La fusion de classifieurs utilise des classifieurs individuels en parallèle et leurs sorties sont combinées [66] de manière à obtenir un "consensus de groupe". Pour cette seconde catégorie on distingue trois niveaux de fusion, à savoir la fusion au niveau décisionnel, la fusion au niveau du rang et la fusion au niveau du score.
  - **Fusion au niveau des scores *(Score level)* :**
    La fusion au niveau des scores utilise les résultats de classification des différents classifieurs utilisés (pour une même modalité ou pour plusieurs différentes). Ici ce sont les scores associés aux classes prédites qui sont combinés puis transmis au module de décision. Par conséquent, un processus de normalisation est nécessaire. Différentes techniques existent [67], on peut citer *"MinMax", "Z-Score", "Double Sigmoïde", "Estimateurs tanh", "Estimateurs biweight"* et d'autres. Ce niveau de décision peut être scindé en deux catégories, à savoir combinaison et classification. La première approche consiste à obtenir un scalaire issu de la combinaison des scores normalisés des différents classifieurs. La seconde considère les scores comme de nouvelles caractéristiques d'entrées pour un nouveau problème de classification à deux classes [68] : accepté et refusé. La fusion au niveau score est considérée comme la plus simple à mettre en œuvre et conserve l'information la plus riche à propos des données d'entrée après classification. De ce fait, elle est la plus utilisée dans la littérature.
  - **Fusion au niveau du rang *(Rank level)* :**
    Lorsqu'un système utilise plusieurs classifieurs et que leur sortie dispose de plusieurs classes, il est possible de les trier dans l'ordre décroissant selon leur score de confiance associé à la prédiction, permettant ainsi d'effectuer une fusion au niveau du rang de la prédiction. Pour ce type de fusion, on peut distinguer trois techniques permettant la combinaison de ces rangs [69] : *"Highest Rank Method", "Borda Count Method"* et *"Régression logistique"*.
    Cette première méthode utilise les scores des classes de chaque classifieur, triées comme spécifié précédemment. À chaque classe est ensuite attribué un unique score correspondant au score du meilleur rang (rang 1 en ordre décroissant) de cette classe parmi les prédictions des classifieurs. Ainsi, on établit un classement général entre les classes, leur rang étant déterminé par le meilleur score attribué à chacune, et la décision finale est prise selon les rangs combinés.

La seconde méthode calcule, pour chaque classe, un rang général combinant la prédiction de chaque classifieur. Ce rang général est issu de la somme du rang de la classe en question en sortie de chaque classifieur. Pour un problème à deux classes, on peut la considérer comme équivalente à un vote à la majorité. Cette méthode ne nécessite aucun apprentissage supplémentaire, cependant, elle ne fait aucune distinction entre les potentielles aptitudes des classifieurs, chacun étant considéré de manière équivalente dans le calcul.
La troisième technique est une généralisation de la méthode précédente (*"Borda Count"*). Une somme pondérée des rangs individuels est calculée et ces poids sont déterminés par régression logistique.

— **Fusion au niveau décisionnel *(Decision level)* :**
Ici, chaque modalité est traitée de manière indépendante par le ou les classifieurs dédiés. Contrairement aux niveaux précédents, le score et le rang ne sont pas considérés. Seule la décision de chaque classifieur (la classe correspondant au meilleur score) est prise en compte et la décision finale est définie en se basant sur la fusion de la classe en sortie de chaque classifieur. Ces décisions peuvent être consolidées par des méthodes telles que le *"vote à la majorité"* [70], le *"Behavior Knowledge Space"* [71], le *"Weighted voting"* [72] et d'autres. Ce niveau de fusion est donc le moins riche en informations et se trouve par conséquent être le moins "puissant" et le plus limité.

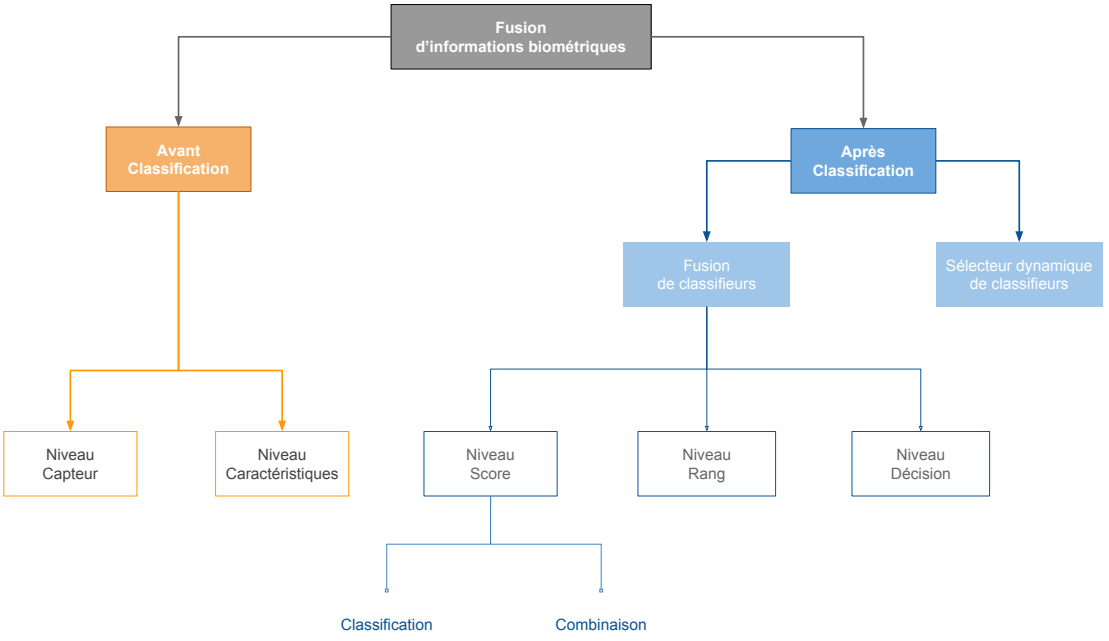



FIGURE 2.18 – Schéma de répartition des différents niveaux de fusion biométrique [59]

### 2.2.2/ Synthèse et solutions proposées

D'une manière générale, les systèmes biométriques faisant appel à la vascularisation, sont particulièrement efficaces et sont considérés, à l'heure actuelle, comme les solutions à modalité unique les plus sures de l'état de l'art. Cependant, il a été montré [59] que les meilleures performances d'authentification sont obtenues pour les différentes combinaisons deux à deux entre l'iris, la paume de la main et les empreintes digitales, issus soit de la fusion au niveau caractéristiques [73], au niveau des scores [74] ou de la décision [75]. Ces modalités correspondent à différentes zones du corps humain et nécessitent, comme spécifié précédemment, une normalisation des caractéristiques pour la fusion à ce niveau. Les empreintes digitales et le réseau vasculaire du doigt appartiennent à la même zone du corps humain, c'est-à-dire le doigt, et sont physiquement comparables (ils présentent des caractéristiques différentes mais ces modalités sont toutes deux formées de lignes courbes). Par conséquent, la fusion de ces modalités semble être facilitée au niveau caractéristiques et les prédictions issues de leur fusion au niveau score présentent de bons résultats d'authentification [76].

En considérant les différentes modalités présentées dans ce chapitre, nous proposons d'orienter notre recherche vers un système multimodal, basé sur la vision pour l'acquisition de ces modalités. La première modalité que nous considérons est le visage. Elle est une des plus acceptées du public, mature et parmi les plus simples à mettre en place. Avec une mesure sans contact et une reconnaissance en temps réel, elle ne nécessite pas de capteurs sophistiqués. En effet, restreinte à une mesure en deux dimensions, une simple caméra de type "webcam" suffit pour acquérir l'image du visage d'un sujet. En revanche, nous avons vu que la modalité du visage en 2D présente un haut taux de contournement, sa falsification étant aisée de par la facilité d'acquisition de cette modalité à l'insu du sujet et de son utilisation par le biais d'une simple image. Une première solution permettant de s'affranchir de ces faiblesses réside dans l'utilisation de caractéristiques en trois dimensions du visage, de par diverses acquisitions sous différents angles par plusieurs capteurs, ou une acquisition en trois dimensions avec un projecteur infrarouge à lumière structurée. Cependant, cette solution requiert une infrastructure conséquente et des besoins calculatoires élevés. Nous avons donc choisi de considérer de nouvelles modalités, sur une zone différente, afin de compléter la mesure du visage.
Les empreintes digitales et le réseau vasculaire du doigt semblent, à nos yeux, représenter une alternative intéressante à l'utilisation de nouveaux capteurs pour le visage. Comme présentés précédemment, les avantages de l'apport de plusieurs modalités dans un même système de sécurité biométrique ne sont plus à démontrer. Ainsi, l'apport d'une modalité déjà commune à de nombreux systèmes et fournissant un haut niveau de sécurité, à savoir les empreintes digitales, couplée à une modalité sous-cutanée et donc plus difficilement "falsifiable", à savoir le réseau vasculaire, représentent un atout non négligeable en matière de fiabilité et de sécurité. Nous proposons donc une mise en œuvre ces deux modalités. Celles-ci seront traitées indépendant, puis fusionnés à la mesure du visage afin d'accroitre le niveau de sécurité. Finalement, dans le but de permettre la mesure de ces deux dernières modalités, nous proposons un système original d'acquisition rotatif mettant en œuvre une caméra permettant un profilage de ces modalités sur le tour du doigt. Ce système d'acquisition permet donc d'obtenir une mesure de mêmes dimensions, sur la même zone du corps, pour chacune de ces modalités. Une potentielle fusion au niveau capteur ou au niveau caractéristiques peut alors être envisagée du fait de leur réelle superposition "physique". Un tel système avec une fusion de ces deux modalités au niveau caractéristiques représente une alternative intéressante aux systèmes hybrides existant.

# 3

# Machine Learning pour la classification d'images

Dans le but de faciliter la compréhension du fonctionnement des méthodes employées, il est question ici d'introduire de manière générale l'apprentissage automatique (*Machine Learning*) pour la classification d'images. En premier lieu, il s'agit de traiter du *Machine Learning* "classique", représentant une partie aujourd'hui maitrisée de l'intelligence artificielle dans la littérature. Puis, en constant développement ces dernières années, une branche plus récente du Machine Learning est présentée, à savoir l'apprentissage profond (*Deep Learning*) et plus précisément les réseaux de convolutions neuronaux (CNN).
Afin d'évaluer les méthodes de classification utilisées dans les chapitres suivants, différentes métriques ou critères statistiques sont introduits. L'axe de recherche de cette étude est ensuite défini plus précisément, grâce aux diverses informations présentées dans cette première partie du manuscrit. La problématique de réduction de la quantité d'informations biométrique à conserver est approfondie (un point rarement étudié dans l'état de l'art) et les modalités biométriques ainsi que les méthodes de classification sont sélectionnées afin de correspondre à notre contexte.

## Sommaire

## 3.1/ Avant-propos

Comme décrit par la figure 3.1, l'apprentissage automatique *(Machine Learning)* fait partie de l'intelligence artificielle et permet, avec l'aide des machines, d'effectuer des calculs afin de résoudre un problème complexe. La particularité du Machine Learning, est que les méthodes utilisées permettent à la machine d'apprendre à réaliser une tâche en fonction de nombreuses données d'entrée. Ainsi, l'algorithme ne se contente pas d'appliquer une consigne définie par son concepteur, mais s'adapte aux données qui lui sont transmise afin d'apprendre comment répondre à la problématique qui lui est fournie. L'apprentissage profond *(Deep Learning)*, que nous traitons également dans ce manuscrit, est quant à lui un sous ensemble du Machine Learning. Le Deep Learning met en œuvre, sous forme de couches, un enchaînement de traitements algorithmiques propres au Machine Learning afin de répondre à un problème complexe découpé en plusieurs tâches, chaque couche utilisant la sortie de la couche précédente comme données d'entrée.

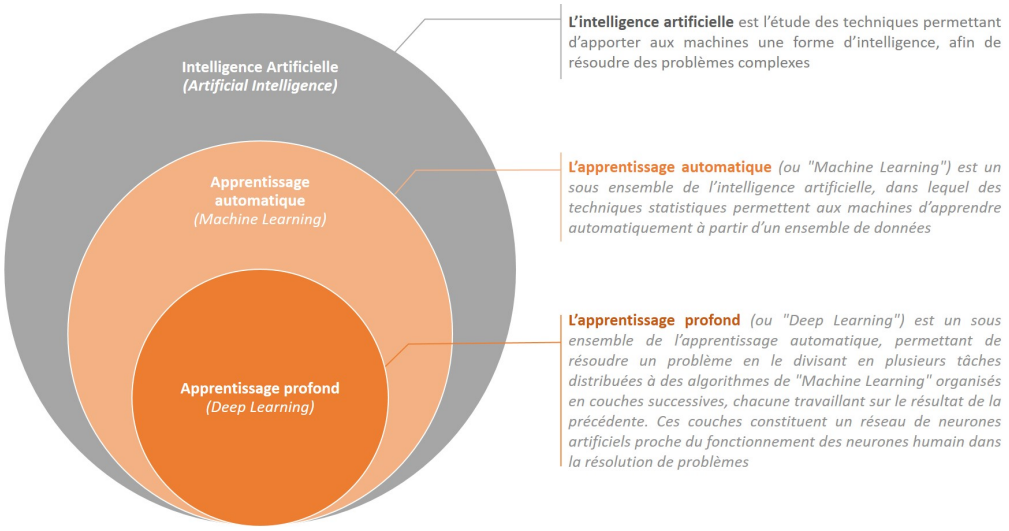



Figure 3.1 – Relations entre les ensembles "intelligence artificielle", "apprentissage automatique" et "apprentissage profond"

De manière générale, un algorithme d'apprentissage automatique utilise un certain nombre de données d'entrées (pouvant être nécessairement très important pour de Deep Learning), représentant des données liées au problème à résoudre. Cet apprentissage dépend de la manière dont sont transmises ces données à l'algorithme de traitement. Il existe donc différentes manières de procéder, et on distingue trois grands groupes d'apprentissage : l'apprentissage supervisée *(supervised learning)*, l'apprentissage non supervisée *(unsupervised learning)*, et l'apprentissage par renforcement *(reinforcement learning)*.

— **Apprentissage supervisé :**
Dans le cas de l'apprentissage supervisé, les données d'apprentissage fournies à l'algorithme contiennent également les solutions associées, que l'on appelle des "étiquettes" ou des "labels". Toutes ces informations sont fournies par un opérateur humain. L'objectif de l'algorithme est alors trouver les relations ($f$) entre les données d'entrée ($X$) et la sortie ($Y$). Ainsi, la relation entre les données d'entrées et la sortie est définie par $Y = f(X)$. L'apprentissage permet donc d'approximer cette fonction afin d'être capable, après l'apprentissage, de prédire une sortie appropriée pour une nouvelle entrée correspondant aux données d'apprentissage (sans label cette fois-ci).

— **Apprentissage non-supervisé :**
Ici, contrairement à l'apprentissage supervisé, aucune donnée d'entrée ne porte de label. Avec l'apprentissage non-supervisé, on laisse l'algorithme découvrir les relations entre les données d'entrée et leur structure. Le système cherche donc à découvrir et apprendre tout ce qu'il est capable de détecter (relations et différences) à partir de ces informations d'entrée et la sortie dépend donc de ce que l'algorithme a su décrire.
— **Apprentissage par renforcement :**
L'apprentissage par renforcement tire son nom du fait que l'apprentissage est amélioré à chaque itération du processus par un retour d'expérience. Les performances de l'algorithme sont maximisées à mesure qu'il prend des décisions à partir de ses données d'apprentissage. L'algorithme est plongé au sein d'un environnement, et prend ses décisions en fonction de son état courant. Après chaque étape d'apprentissage, l'environnement retourne une récompense, qui peut être positive ou négative, en fonction du résultat de l'étape précédente. Ainsi, au travers d'expériences itérées, l'algorithme cherche un comportement décisionnel optimal, afin qu'il maximise la somme des récompenses au cours du temps.

On peut également citer l'apprentissage semi-supervisé *(semi-supervised learning)*. Cependant, ce n'est pas un groupe à part entière, du fait qu'il dérive à la fois de l'apprentissage supervisé et non supervisé. Certaines données portent une étiquette mais la majorité demeure non annotées. L'acquisition de données annotées pour l'apprentissage requiert des compétences humaines ainsi que du temps, tandis que le coût associé à l'acquisition de données non annotées est très court en comparaison. Ainsi, les algorithmes d'apprentissages semi-supervisés cumulent les avantages des deux méthodes, permettant, avec la faible quantité de données annotées, d'orienter l'apprentissage, puis, avec les données non annotées, de découvrir une plus large quantité d'informations relationnelles entre les données.
Dans le cadre de l'apprentissage automatique, on peut établir plusieurs groupes de problématiques, auxquels les types d'apprentissages peuvent répondre. On retrouve principalement les problématiques de "classification", de "régression", d'"associations de règles" et de "clustering", que nous explicitons ci-après. Certains algorithmes se prêtent mieux à un type d'apprentissage particulier et à certaines problématiques mais ils n'y sont pas limités. Dans certains cas, afin d'utiliser un algorithme supervisé pour la classification de données non annotées, il est possible d'utiliser en complément un algorithme non supervisé dans le but de séparer les données de manière automatique. Le choix de l'algorithme dépend alors de nombreux facteurs tels que la quantité de données, leur qualité et leur nature, la puissance de calcul disponible ainsi que les temps de calculs acceptables selon l'application, etc.

La classification a pour objectif d'assigner des données à des catégories définies au préalable. Lors de l'apprentissage, les relations entre les données et leurs catégories (ou classes) sont déterminées et apprises. Ainsi, la classification permet d'obtenir, pour un nouvel échantillon, une prédiction d'appartenance à une classe connue, telle que "malade"/"sain" et "rouge"/"bleu"/.../"vert". Dans le cas d'un problème à deux classes on parle de classification binaire ou binomiale, et dans le cas à plus de deux classes, on parle de classification à classes multiples ou multiclasses.
La régression a pour but de prédire la valeur d'une certaine quantité à l'évolution continue, dépendant de certaines caractéristiques. Ce problème est établi lorsque la donnée de sortie est une valeur réelle, telle qu'une somme d'argent ou un poids. On peut citer l'estimation de valeurs telles que l'évolution des prix de locations d'appartement selon le secteur et les caractéristiques du bien, ou bien la durée de vie humaine en fonction des conditions de vies et d'autres critères.

Un problème d'apprentissage d'association de règles se présente comme un problème où l'on souhaite associer des règles décrivant majoritairement des grandes parties des données (e.g les personnes qui achètent un objet X ont également tendance à acheter un objet Y). Une règle d'association comporte deux parties : un antécédent ("si") et un résultant ("alors"). Un antécédent est un élément compris dans les données, et un résultant est une combinaison d'antécédent suivant une relation précise. Ainsi, l'algorithme recherche au travers des données la fréquence de modèles "si-alors" afin d'identifier les relations les plus importantes entre les données.
Le clustering se présente comme un problème où l'on souhaite établir des groupes (ou des "clusters") selon des caractéristiques particulières propres aux données (e.g regrouper des clients selon leurs habitudes et leur comportement d'achat). Comme exemple particulier au clustering on peut citer la détection d'anomalies dont l'objectif est d'identifier les points de données qui sont inhabituels. Dans certains cas, les variations possibles sont si nombreuses et les exemples réels d'apprentissage si rares, qu'il n'est pas possible de savoir à quoi ressemble une activité frauduleuse. L'approche de la détection des anomalies consiste donc simplement à apprendre à quoi ressemble l'activité normale et d'identifier tout ce qui est très différent.

Pour la suite de ce manuscrit, nous allons nous intéresser à l'apprentissage supervisé dans le cadre de la classification d'images. Du fait de notre problématique, à savoir déterminer si une personne est autorisée ou non autorisée, nous nous focaliserons sur une étude de classification binaire. Avant de considérer une quelconque application à la biométrie, nous allons établir les généralités propres au traitement d'image et communes à toutes les applications de ce domaine.
Comme le montre la figure 3.2, la chaine de traitement générale (commune aux algorithmes de classification d'images) comporte plusieurs étapes clés telles que la détection et l'isolation de zone d'intérêt *(Region of Interest - ROI)*, des prétraitements afin d'améliorer l'image dans le sens voulu suivi d'une potentielle normalisation, l'extraction de caractéristiques ainsi que leur sélection, puis la classification effectuant soit un apprentissage d'un modèle, soit une prédiction à partir du modèle appris. Cette prédiction donne lieu à une décision qui, à partir de multiples résultats moyennés sur une base de données d'images, fourni un certain nombre d'informations servant à évaluer les performances du modèle.

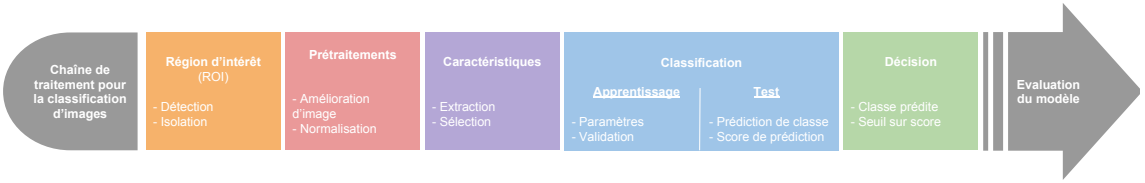



FIGURE 3.2 – Chaîne des étapes de classification d'images communes à la majorité des applications

La première étape consiste à extraire la zone d'intérêt dans laquelle se trouve les éléments que l'on veut classifier. Cette étape, bien que facultative, permet de réduire l'influence des données peu pertinentes dans l'image. Par exemple, dans le cas de la reconnaissance de visages, il est important d'isoler le visage du fond de l'image. En revanche, dans le cas d'une détection multiple (piétons, voiture, panneaux, etc.), les informations sont réparties dans toute l'image.
Les prétraitements constituent une étape non négligeable de cette chaîne. La manipulation de l'image (filtres, segmentation binaire, superpixels, changement d'espaces couleurs, etc.) permet de renforcer ou faire apparaître les éléments importants tout en réduisant les contributions non utiles. La phase de normalisation permet de conserver une homogénéité entre les images utilisées pour l'apprentissage. Les images sont redimensionnées à une même dimension et une rotation de l'image peut être envisagée pour conserver une orientation similaire du sujet à observer.

L'extraction de caractéristiques est une étape cruciale, permettant de transcrire l'information visuelle en une information plus stable et compréhensible par l'algorithme d'apprentissage. Certains algorithmes, basés sur la répartition statistique des données, permettent de trouver les caractéristiques communes aux images d'une simple classe ou bien de détecter les caractéristiques discriminant au mieux une classe par rapport à une autre tout en cherchant à réduire l'impact des caractéristiques communes à plusieurs classes. Ces caractéristiques ne représentent pas toutes la même contribution dans l'image et leur nombre peut rapidement être important. C'est pourquoi une phase complémentaire de sélection de caractéristiques permet de limiter leur nombre tout en ne conservant que les plus pertinentes.

La classification est divisée en deux phases : l'apprentissage et le test (ou l'utilisation). Lors de l'apprentissage, les caractéristiques des images d'apprentissages sont utilisées par le classifieur afin de construire un modèle à partir de règles séparant au mieux les classes. Selon l'application et les images utilisées, il est nécessaire d'ajuster les paramètres de classification, et l'apprentissage peut être perfectionné en utilisant une technique de validation croisée *(ou "cross validation")*, permettant de scinder les images en plusieurs groupes et d'effectuer un apprentissage sur chacun de ces groupes en faisant varier les paramètres.
La phase d'utilisation permet, à partir des caractéristiques d'une image inconnue, de déterminer l'appartenance de l'échantillon à une certaine classe en utilisant le modèle de classification. Ainsi ce modèle fourni l'étiquette de la classe prédite et le score associé attestant de la probabilité d'une bonne prédiction. L'étape de décision consiste à définir si la prédiction du modèle est juste ou non. Soit la classe prédite est utilisée directement (prédiction juste), soit, un seuil peut être défini et ajusté sur le score de la prédiction afin de considérer une prédiction comme juste seulement si le score est supérieur au seuil fixé. Pour finir, l'évaluation permet de quantifier les performances du modèle avec des métriques calculées à partir d'informations issues des résultats de prédiction sur des images de test. Ces métriques permettent de comparer des modèles entre eux et ainsi sélectionner le modèle optimal dans un cas donné.

Lors de l'apprentissage, une mauvaise répartition des données d'entrées ainsi qu'un mauvais paramétrage de la classification peuvent mener à un "sur-apprentissage" ou à un "sous-apprentissage". Le sur-apprentissage correspond à un modèle considérant les détails et le bruit comme des données importantes, menant à d'excellentes performances lors de l'apprentissage, mais de très mauvaises sur de nouvelles images inconnues. Le sous-apprentissage fait référence à un modèle incapable de fournir de bonnes performances en apprentissage ni à partir de nouvelles données.

## 3.2/ Machine Learning "classique"

Comme décrit précédemment, le Machine Learning est une sous-famille de l'intelligence artificielle. Ce que nous considérons comme Machine Learning "classique", ou "traditionnel", correspond à cette famille à laquelle on exclut la sous-famille du Deep Learning. Ainsi, le Machine Learning classique correspond aux algorithmes et aux méthodes désormais matures de l'état de l'art et ayant d'ores et déjà fait leurs preuves sur de nombreuses problématiques.

### 3.2.1/ Extraction et traitement des caractéristiques

L'extraction de caractéristiques est une étape cruciale de la chaîne de traitement de la classification d'images, et les méthodes utilisées dans la littérature peuvent être regroupées selon deux catégories principales [77], comme le montre la figure 3.3, à savoir les méthodes globales et les méthodes locales. Les méthodes globales sont indépendantes de l'application (du type d'élément présent

dans l'image) et se focalisent sur des caractéristiques telles que les textures, la couleur ou encore les formes. On peut considérer trois niveaux d'abstraction pour ces caractéristiques : le niveau pixel (couleur, coordonnées dans l'image, etc.), le niveau local (caractéristiques issues de subdivision de l'image, de segmentation, ou autre) et le niveau global (caractéristiques issues de l'image entière ou d'une seule région de l'image). En comparaison, les méthodes locales sont dépendantes de l'application et représentent souvent une synthèse de caractéristiques bas-niveau. Chacune de ces deux catégories peuvent être sous divisées selon le type de techniques [78].

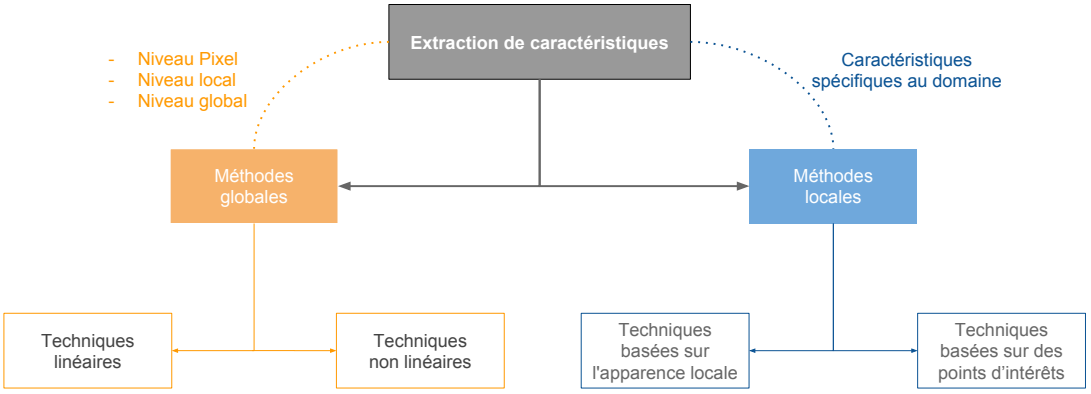



FIGURE 3.3 – Catégories de méthodes d'extractions de caractéristiques dans l'image

— **Méthodes globales :**
Les approches globales, également appelées méthodes basées sur l'apparence *(Appearance-based methods)*, utilisent l'image complète. De manière générale, l'image est représentée selon une matrice de pixels, qui est transformée en un vecteur de pixels pour faciliter les manipulations. Ces approches sont sensibles aux variations d'acquisition (orientation, éclairage, etc.) du fait que le moindre changement induit une variation de la valeur des pixels. Ces méthodes utilisent un sous espace (de faible dimension) du sujet à considérer, afin d'y projeter de nouvelles images et ainsi réduire l'influence d'éléments non pertinents dans l'image. Considérant les techniques utilisées pour générer cette projection, cette catégorie regroupe deux types de techniques : les linéaires et les non linéaires.
Les techniques linéaires effectuent une projection linéaire des données d'entrées, représentées dans un espace de très grande dimension (dépendant du nombre de pixels et de canaux de l'image), dans un nouvel espace relativement réduit et dédié au "sujet". Cet espace est composé de caractéristiques propres aux types d'éléments observés. De nombreuses techniques peuvent être classées comme linéaires : l'Analyse en Composantes Principales "ACP" *(ou "PCA" pour "Principle Component Analysis")*, l'Analyse en Composantes Indépendantes "ACI" *(ou "ICA" pour "Independant Component Analysis")*, l'Analyse Discriminante Linéaire "ADL" *(ou "LDA" pour "Linear Discriminent Analysis")*, les ondelettes de Gabor, et beaucoup d'autres. L'ACP [79, 80, 81] est une approche très populaire, sur laquelle nous nous attarderons plus particulièrement, de fait de son importante utilisation par de nombreux algorithmes dans le domaine de la biométrie. L'ACI [82, 83] est une méthode issue d'applications statistiques et de traitement du signal, spécialement utilisée pour effectuer une séparation "aveugle" de sources. l'ACI consiste à exprimer un ensemble de N variables aléatoires $x_1, ..., x_N$ comme une combinaison linéaire de N variables aléatoires $s_j$ statistiquement indépendantes. L'ADL [84], contrairement à l'ACP qui construit un sous

espace pour décrire de manière optimale un type "d'objet", construit un sous espace discriminant pour les distinguer, de manière optimale, les variations entre différentes catégories d'un même type d'objet. En d'autres termes, l'ADL permet de trouver des caractéristiques séparant au mieux les classes.
Lorsque la structure des données d'entrée n'est pas linéaire, une solution consiste à utiliser une fonction nommée fonction noyau *(ou kernel)*. Dans ce cas, un espace de grande dimension est créé dans lequel la représentation du problème devient linéaire. Les méthodes linéaires peuvent être adaptées afin de répondre à ce problème, de par l'ajout d'un noyau [85]. Ainsi, on retrouve l'Analyse en Composantes Principales avec Noyau *(Kernel PCA - KPCA)* [86], l'Analyse en Composantes Indépendantes avec Noyau *(Kernel ICA - KICA)* [87], Analyse Discriminante Exponentielle "ADE" *(ou EDA - "Exponential Discriminant Analysis")* [88], et nombreux autres.

— **Méthodes locales :**
Les méthodes locales, ne traitant que des caractéristiques très spécifiques au type d'informations observées dans l'image, peuvent être divisées en deux catégories : les techniques basées sur l'apparence locale et les techniques basées sur des points d'intérêts.
Les techniques basées sur l'apparence locale divisent l'image en petites régions ou "patchs" desquelles des caractéristiques locales sont directement extraites. Une fois ces régions définies, il advient de choisir la meilleure façon de représenter l'information de chaque région. Les caractéristiques les plus généralement utilisées dans la littérature sont les coefficients de Gabor [89], les ondelettes de Haar [90], la transformation de caractéristiques visuelles invariante à l'échelle *(Scale-Invariant Feature Transform - SIFT)* [91], les caractéristiques basées sur les motifs binaires locaux *(Local Binary Pattern - LBP)* [92] et leurs extensions tels que les motifs ternaires locaux *(Local Ternary Pattern - LTP)* [93], et d'autres.
Les techniques basées sur des points d'intérêts détectent d'abord ces points spécifiques, permettant par la suite d'extraire les caractéristiques représentant les diverses relations entre ces points tels que leur distance les uns avec les autres, leur angle, etc. Ce type d'extraction est réalisé avec des approches telles que l'Architecture de Liens Dynamiques *(Dynamic Link Architecture - DLA)* [94], l'extraction par filtre de Gabor [95], etc.

Une troisième catégorie pourrait être définie, composée de méthodes hybrides combinant des techniques des deux catégories précédentes et de nouvelles méthodes basées sur des modèles statistiques. Pour une compréhension plus fine de la suite de ce manuscrit, nous proposons de détailler l'algorithme de l'ACP en annexe A.1.1. L'objectif de cette méthode, dans un contexte d'analyse d'images, consiste à exprimer un ensemble de M images selon une base de vecteurs orthogonaux particuliers (des vecteurs propres), contenant des informations indépendantes d'un vecteur à l'autre. Ainsi, ces vecteurs sont une représentation simplifiée et optimale de ces images dans le sens où la variance est maximisée. Chaque vecteur décrit alors des caractéristiques indépendantes. Une projection d'une nouvelle image dans cette base permet de représenter l'information de manière normalisée, selon une combinaison linéaire de ces vecteurs propres, et donc de faciliter la comparaison de ces nouvelles données.

### 3.2.2/ Classification

L'extraction de caractéristiques fournit donc un vecteur composé d'éléments représentant ces caractéristiques. L'étape suivante de la chaîne est la classification. Son but est de calculer le degré similitude entre 2 vecteurs (caractéristiques cible et caractéristique mesurée) ou entre 1 vecteur (caractéristique mesurée) et un ensemble de vecteurs (formant une classe). Cette comparaison peut être effectuée de différente façons, plus ou moins efficaces selon la complexité des données (dimensions des vecteurs, variance interclasse, séparation des classes, etc.).

#### 3.2.2.1/ Calculs de distances

De manière générale, la façon la plus triviale de comparer deux vecteurs d'une même caractéristique (et donc de même dimension) est de vérifier leur degré de similitude. En statistique, cette similitude est exprimée comme étant la distance séparant ces deux vecteurs dans leur espace. De la même manière que pour mesurer la distance entre deux points (norme), le calcul de distance entre deux vecteurs mesure la distance entre chaque élément "i" de ces vecteurs.

Soient deux vecteurs "$x$" (caractéristiques cibles) et "$y$" (caractéristiques mesurées) de "$n$" éléments, appartenant à un espace vectoriel normé $E$, tels que $\vec{x}(x_1, ..., x_i, ..., x_n)$ et $\vec{y}(y_1, ..., y_i, ..., y_n)$ sont tous deux éléments de $\mathbb{R}^n$. La distance entre ces vecteurs, notée $d(\vec{x}, \vec{y})$, est une mesure entre chacune de leur composante "i" deux à deux ($1 < i < n$) et peut se définir de plusieurs manières :

— **Distance de Minkowski (p-distance ou Lp) :**
Généralisation du calcul de distance entre deux vecteurs à l'ordre "p".

$$d(\vec{x}, \vec{y}) = \sqrt[p]{\sum_{i=1}^{n} |x_i - y_i|^p} \tag{3.1}$$

— **Distance de Tchebychev ($\infty$-distance ou $L_\infty$) :**
C'est la distance, entre deux vecteurs, donnée par la différence maximale entre leurs coordonnées. Elle est équivalente à la distance de Minkowski d'ordre infini ($p = \infty$).

$$d(\vec{x}, \vec{y}) = \lim_{p \to \infty} \sqrt[p]{\sum_{i=1}^{n} |x_i - y_i|^p} = \sup_{1 \leq i \leq n} |x_i - y_i| \tag{3.2}$$

— **Distance Euclidienne (2-distance ou L2) :**
La distance Euclidienne est la plus courte distance entre deux vecteurs, également appelée distance à vol d'oiseau. Elle dérive de la distance de Minkowski à l'ordre 2 ($p = 2$).

$$d(\vec{x}, \vec{y}) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2} \tag{3.3}$$

— **Distance de Manhatan (1-distance ou L1) :**
Elle représente la distance parcourue entre deux points ou vecteurs sur une "grille" pouvant s'apparenter aux rues d'une ville et est donc appelée "taxi-distance". Elle dérive de la distance de Minkowski à l'ordre 1 ($p = 1$).

$$d(\vec{x}, \vec{y}) = \sum_{i=1}^{n} |x_i - y_i| \tag{3.4}$$

— **Distance de Mahalanobis :**
Elle diffère de la distance euclidienne par le fait qu'elle prend en compte la variance et la corrélation de la série de données. Ainsi, à la différence de la distance euclidienne où toutes les composantes des vecteurs sont traitées indépendamment et de la même façon, la distance de Mahalanobis accorde un poids moins important aux composantes les plus dispersées.

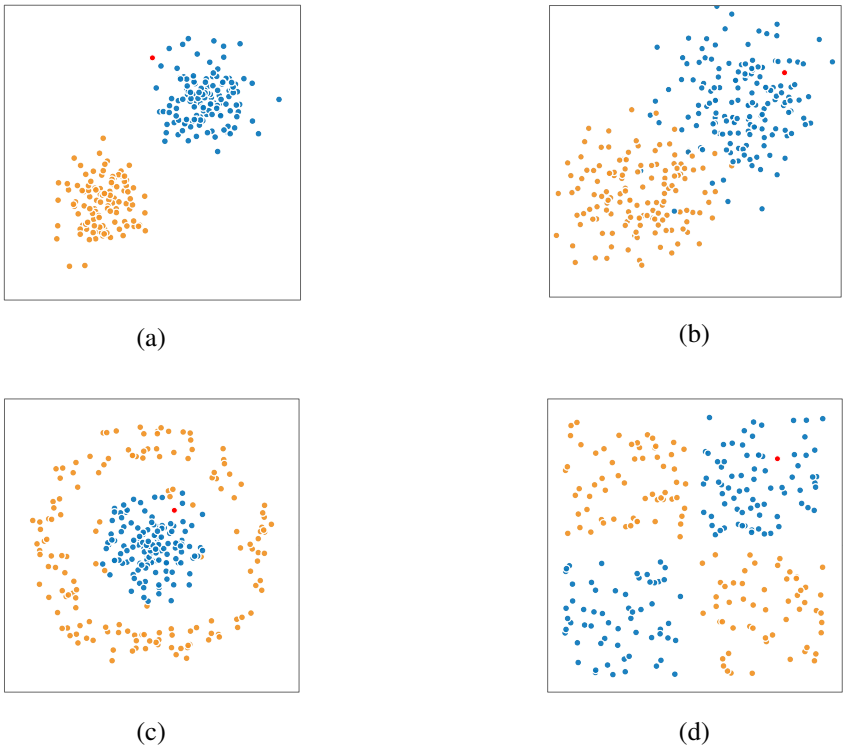
$$d(\vec{x}, \vec{y}) = \sqrt{(\vec{x} - \vec{y})^T \Sigma^{-1} (\vec{x} - \vec{y})} \tag{3.5}$$

où $\Sigma^{-1}$ est l'inverse de la matrice de covariance entre les vecteurs $\vec{x}$ et $\vec{y}$

D'autres mesures de distances peuvent être citées, telles que la distance de Camberra, la distance $\chi^2$, la distance similarité cosinus, ou encore la distance de Hausdorff, etc. La distance ainsi mesurée donne une information concernant la relation entre le vecteur de caractéristiques enregistrées, et le vecteur de caractéristiques mesurées. La correspondance entre les deux vecteurs est donc à définir selon un seuil sur cette valeur de distance. Le type de distance à choisir, chacune fournissant diverses informations, dépend des données composant les caractéristiques (dimensions, complexité, etc.), du type d'application et donc de l'extracteur de caractéristiques utilisé. En effet, Charu C. Aggarwal et al. [96] ont montré que, bien que très peu d'attention ne soit portée à cette relation entre le type de données et le type de mesure de distance, le comportement de certaines métriques d'ordres proches de 1 voir fractionnels produisent des résultats inattendus sur des données de très grande dimension., et sont parfois préférables aux mesures standards.

#### 3.2.2.2/ Méthodes "statistiques"

Selon Taiwo Oladipupo Ayodele [97], les méthodes supervisées du Machine Learning appliquées à un problème de classification peuvent principalement se résumer au travers des ensembles de types suivants : les classifieurs linéaires, les classifieurs quadratiques, les regroupements par K-moyennes, le *boosting*, les arbres de décision, les réseaux de neurones et les réseaux bayesiens, etc. Parmi les classifieurs linéaires, on retrouve entre autres la régression logique, les perceptrons, les machines à vecteurs supports *(Support Vector Machines - SVM)* ou encore les classifieurs bayesiens naïfs,. Les arbres de décisions aléatoires, si plusieurs sont utilisés simultanément et initialisés à partir de différentes caractéristiques, forment les forêts aléatoires *(Random Forest - RF)*.



(a) (b) (c) (d)

Figure 3.4 – Exemples de répartitions de deux classes de données dans un plan 2D - (a) représente des données linéairement séparables ; (b) représente des données non linéairement séparables pouvant tout de même être séparées par une droite en tolérant des erreurs ; (c) et (d) représentent des données non linéairement séparables

Le choix des méthodes se fait selon le type, la répartition et la dimension des données. Une comparaison est effectuée dans le tableau 3.1. En effet, comme le montre figure 3.4 de par des exemples simples (deux classes représentées par les couleurs orange et bleu, un nouvel échantillon inconnu en rouge), les données peuvent être linéairement séparables ou non linéairement séparables. La figure 3.4a représente des données linéairement séparables. En effet, la répartition des données dans ce plan en deux dimensions suggère qu'elles peuvent être séparées par une simple droite (ou un hyperplan dans la cas d'une dimension supérieure à deux). La figure 3.4b se trouve ne pas être linéairement séparable car, comme on le remarque, les données se rejoignent. Cependant le barycentre des classes demeure inchangé, la même droite peut alors être considérée pour séparer les données, faisant alors apparaître des erreurs de classification dues au bruit. Les figures 3.4c et 3.4d représentent des cas de données non linéairement séparables. Sur la première, les données sont toujours séparées en deux classes distinctes, mais non linéairement séparables par un plan. La deuxième figure représente également deux classes distinctes, mais cette répartition nécessite une séparation par deux droites ou deux hyperplans et ne peut donc pas être considérée comme linéairement séparable.

Nous proposons de présenter des méthodes précédemment citées :

— **Classifieurs linéaires :**
L'objectif des modèles linéaires, dans le cas de la classification à deux classes, est de séparer l'espace des données d'entrées par un hyperplan, représentant une frontière de décision. Cette classification permet de grouper les données selon leurs caractéristiques similaires et donc leur répartition dans leur espace vectoriel. Ainsi, chaque côté de cet hyperplan représente une classe, contenant la majorité ou toutes les données d'apprentissage, selon le niveau de bruit présent dans la répartition de ces données. Par conséquent, un nouvel échantillon est défini comme appartenant à une certaine classe selon qu'il se trouve d'un côté ou de l'autre de l'hyperplan. Les classifieurs linéaires sont souvent utilisés dans des applications où le temps de calcul représente une forte contrainte. Ils sont en effet réputés pour figurer parmi les classifieurs les plus rapides [97].
La régression logique [98] permet d'estimer les paramètres d'un modèle logistique, lequel est un modèle de classification se servant d'une fonction logistique (e.g fonction sigmoïde) pour modéliser les dépendances entre des données issues de deux classes. Ce modèle possède une variable dépendante aux données d'apprentissage, prenant deux valeurs possibles (0 ou 1) illustrant l'état des données par rapport aux classes (vainqueur/perdant, sain/-malade, etc.). La probabilité d'appartenance à une certaine classe est déterminée par le "log-odds" (logarithme des probabilités), formulant une combinaison linéaire de variables indépendantes (prédicteurs), lesquelles pouvant être binaire (0 ou 1) ou continue. La fonction logistique se charge ensuite de convertir ce log-odds en probabilité variant alors de 0 à 1 (respectivement forte probabilité d'appartenance à la classe 0 ou la classe 1).
Un perceptron [99] est un classifieur binaire correspondant à un neurone sous sa forme la plus simple. Muni d'une règle d'apprentissage permettant de déterminer automatiquement les poids du neurone, il dispose d'une seule sortie. Un perceptron multicouche est un ensemble de perceptrons, organisés en plusieurs couches constituées de plusieurs neurones et de plusieurs sorties. Ces perceptrons ne contiennent aucune boucle, il s'agit d'une propagation directe de l'information.
Les Machines à Vecteurs Supports (SVM)[100] effectuent une classification par la construction d'un hyperplan de $N$ dimensions, séparant de manière optimale les données d'entrées, elles aussi de dimensions N, en deux catégories. Les SVM introduisent la notion de marge, représentant la distance entre les données d'apprentissage et l'hyperplan. L'objectif est alors de trouver un hyperplan maximisant cette marge, réduisant ainsi l'erreur à un

minimum. Dans des cas de données non linéairement séparables, les SVM peuvent fournir un ensemble d'hyperplan. Dans des cas plus non linéaires plus complexes où la répartition des données est moins triviale, les SVM peuvent être associés à une fonction noyau (polynomiale, RBF, etc) afin de définir des séparateurs plus complexes. Le fonctionnement des SVM sera explicité plus précisément dans la suite de ce manuscrit.

— **K-moyennes et K-plus proches voisins :**
La première étape du regroupement par les K-moyennes [101] consiste à déterminer le nombre *K* de groupes et de considérer le centre de chacun d'entre eux. Ce centre est défini à priori de manière aléatoire au sein du groupe (apprentissage non supervisé) mais peut être choisi selon certaines données annotées (apprentissage semi-supervisé). Par la suite, l'algorithme effectue en boucle trois étapes jusqu'à la convergence : recalculer les coordonnées du barycentre de chaque ensemble, déterminer la distance entre chaque donnée et ce centre, puis compléter les ensembles en regroupant les échantillons les plus proches.
Les K-plus proches voisins *( k-Nearest Neighbors - KNN)* [102] font, quant à eux, partie de l'apprentissage supervisés. Les K-NN n'ont pas besoin de construire un modèle prédictif, il n'existe pas de phase d'apprentissage proprement dite (ils sont donc souvent associés au terme *"Lazy Learning"*). Pour effectuer une prédiction, l'algorithme se base sur le jeu de données complet et, pour une nouvelle observation, va rechercher les *K* instances du jeu de données les plus proches de l'observation. Ensuite pour ses *K* voisins, l'algorithme se base sur leurs variables de sortie pour assigner le nouvel échantillon à une classe.

— **Arbres de décision et forêts aléatoires :**
Les arbres de décisions [103] sont des "arbres" classifiant des instances de par un tri basé sur les valeurs de leurs caractéristiques. Chaque nœud traite une certaine caractéristique et représente un seuil sur la valeur de cette caractéristique et chaque branche détermine la plage de valeurs prise par cette caractéristique selon la définition du nœud. La caractéristique séparant au mieux les données d'apprentissage est alors définie comme étant le nœud racine de l'arborescence, et chaque nouvelle branche donne lieu à un nouveau nœud traitant une nouvelle caractéristique découlant des valeurs de la caractéristique principale.
L'idée principale définissant les forêts aléatoires *(Random Forest - RF)* [104] est la combinaison de plusieurs arbres de décision en un unique modèle, chaque arbre de décision incorporant sa propre expérience sur une certaine caractéristique. Chacun de ces arbres formant la "forêt" considère un sous-ensemble aléatoire de caractéristiques, n'ayant accès qu'à un sous-ensemble aléatoire de données d'apprentissage. Ainsi, la diversité des caractéristiques traitées est augmentée, produisant une prédiction plus robuste, celle-ci étant issue d'une moyenne des estimations des arbres de décisions. De la même manière que pour les SVM, le fonctionnement des RF sera détaillé plus précisément dans ce manuscrit.

— **Réseaux bayésiens et réseaux de neurones :**
Un réseau bayésien *(Bayesian Network - BN)* [105] est un modèle graphique établissant des relations probabilistes parmi un ensemble de données (des caractéristiques), appelé aussi graphe orienté acylique *(Directed Acylic Graph - DAG)* et peut être vu comme un graphe hiérarchique. Chaque nœud dispose de paramètres probabilistes, sous la forme de distributions locales conditionnelles tenant compte des nœuds parents. La fonctionnalité la plus intéressante des BN, comparé aux arbres de décision et aux réseaux de neurones, est la possibilité de définir la structure du réseau en tenant compte au préalable d'informations sur un problème donné, en terme de relations structurelles entre les caractéristiques.
La particularité d'un réseau de neurones, "RN" *(Neural Networks - NN)* [106], est sa faculté à effectuer simultanément plusieurs tâches de régression et ou de classification. Un réseau de neurones artificiels, "RNA" *(Artificial Neural Network - ANN)*, dépend de trois aspects fondamentaux : les fonctions d'entrée et d'activation, l'architecture du réseau et les poids

de chaque connexion entre les nœuds. Ces poids, d'abord initialisés à des valeurs aléatoires, sont ajustés lors de la phase d'apprentissage de par la propagation répétée de données d'apprentissage au travers du réseau. Les données d'entrée donnent alors lieu à un résultat, dépendant des poids des nœuds, qui est alors comparé au couple formé par la classe désirée (vérité de terrain) et la probabilité associée. Puis les poids sont légèrement modifiés dans le but de fournir un nouveau résultat plus proche de la valeur attendue.

Le tableau 3.1 établit une comparaison rapide des certains algorithmes présentés précédemment, à partir d'une évaluation sur certaines caractéristiques de ces classifieurs. Les performances sur ces évaluations sont représentées par des étoiles "*". Le maximum de quatre étoiles fait référence aux meilleures capacités, tandis que le minimum d'une étoile représente les plus faibles performances. On observe alors que les SVM, les arbres de décision et les réseaux bayésiens sont globalement les plus performants. On peut facilement estimer les capacités des forêts aléatoires comme étant au moins au niveau des SVM, du fait qu'elles sont une combinaison de plusieurs arbres de décision. La publication [107] de laquelle est tirée cette table datant de 2007, elle n'intègre pas de comparaison avec les réseaux de neurones profonds. Les RN représentent donc ici les réseaux de neurones dans leur forme classique.

| | **AD** | **RN** | **RB** | **KNN** | **SVM** |
|---|---|---|---|---|---|
| **Performances de classification** | ** | *** | * | ** | **** |
| **Vitesse d'apprentissage selon le nombre d'éléments** | *** | * | **** | **** | * |
| **Vitesse de classification** | **** | **** | **** | * | **** |
| **Tolérance aux données incomplètes** | *** | * | **** | * | ** |
| **Tolérance aux données erronées** | *** | * | ** | ** | **** |
| **Tolérance aux données redondantes** | ** | ** | * | ** | *** |
| **Tolérance au bruit** | ** | ** | *** | * | ** |

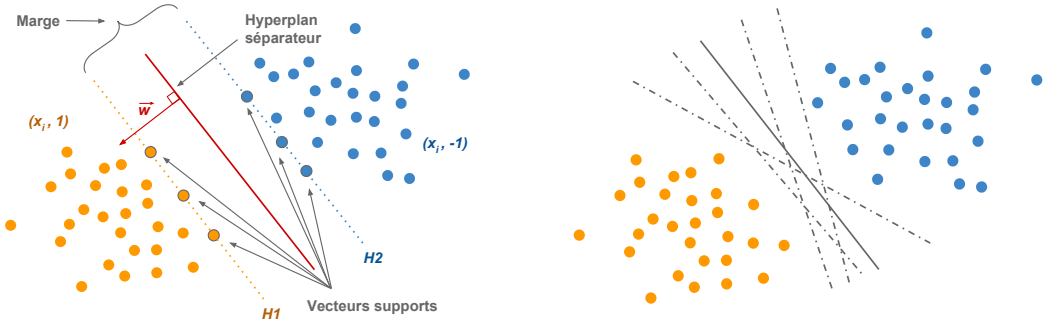
TABLE 3.1 – Comparaison d'algorithmes de classification supervisée [107] (**** représente les meilleures performances ; AD : Arbre de Décision, RN : Réseaux de Neurones peu profond, RB : Réseaux Bayesiens, KNN : Plus proches voisins, SVM : Séparateurs à Vastes Marges )

### Séparateurs à Vastes Marges *(SVM - Support Vector Machine)*

Les Séparateurs à Vastes Marges, encore appelés Machines à Vecteurs Supports (SVM), cherchent à séparer au mieux des ensembles de données appartenant à des classes. Cette séparation, comme illustrée par la figure 3.5a, est effectuée par un hyperplan (ou une droite dans un plan en deux dimensions). Celui-ci est généré, ou supporté, par certains vecteurs appartenant aux classes à séparer, que l'on appelle alors vecteurs supports. Pour chaque classe, un hyperplan "support" passe par ces vecteurs supports et l'hyperplan séparateur est situé à égale distance de chacun d'eux. Cette distance est appelée "la marge". Il existe une infinité de lignes ou d'hyperplans séparant des données linéairement séparables, hors le but est d'effectuer une classification et donc d'être capable de prédire, avec une erreur minimale, l'appartenance d'une nouvelle donnée à telle ou telle classe. Ainsi, l'objectif des SVM est de trouver l'hyperplan maximisant la marge, séparant au mieux les classes et minimisant l'erreur. Comme le montre la figure 3.5b, plusieurs hyperplans sont générés à partir de divers vecteurs supports, et l'hyperplan fournissant la marge maximale est alors retenu.

Une présentation plus complète de l'obtention des hyperplans et de la sélection des vecteurs supports est proposée en annexe A.1.2. Ici, nous illustrons rapidement les différents types de séparateurs fournis par les SVM, selon la répartition des données.

**Séparateurs linéaires - Cas linéairement séparable :**




(a) Principe de la marge des SVM (b) Recherche d'un hyperplan maximisant la marge

FIGURE 3.5 – Séparations de classes par les SVM dans le cas de données linéairement séparables

Considérons un ensemble de $N$ données de dimensions $n$ (c'est à dire $N$ vecteurs chacun de $n$ éléments) nommées $\vec{x_i}$, où $i \in [1; N]$. Ces données sont réparties selon deux classes nommées $y_i$ telles que $y_i \in \{+1, -1\}$ et sont représentés par le couple $(x_i, yi)$. Soient le vecteur $\vec{w}$ représentant la normale de l'hyperplan séparateur, tel que $\vec{w} = (w_1, ..., w_n)$, et $b$ la constante de biais (ordonnée à l'origine). L'hyperplan $H$ séparant ces données, et les hyperplans supports $H_1$ et $H_2$ vérifient l'équation suivante :

$$f(\vec{x_i}) = y_i(\vec{w}.\vec{x_i} + b) \geq 1 \tag{3.6}$$

Cette équation signifie que tous les points d'une même classe sont du même côté de l'hyperplan, le signe de $f(x_i)$ donnant la classe associée du vecteur $x_i$. Ces hyperplans ont alors pour équation :

$$\begin{cases} H_1 & : \vec{w}.\vec{x_i} + b = 1 \\ H & : \vec{w}.\vec{x_i} + b = 0 \\ H_2 & : \vec{w}.\vec{x_i} + b = -1 \end{cases} \tag{3.7}$$

La distance entre un vecteur $\vec{x}$ et l'hyperplan $H$, ainsi que la marge $M$, sont définis par :

$$d(\vec{x}, H) = \frac{f(\vec{x})}{||\vec{w}||} \qquad M = \frac{2}{||\vec{w}||} \tag{3.8}$$

Déterminer l'hyperplan optimal équivaut à maximiser la marge $M$, ce qui revient donc à minimiser $||\vec{w}||$. C'est un problème d'optimisation quadratique, appelé formulation primale, qui peut être résolu grâce à la méthode de Lagrange. Cette méthode permet de sélectionner les vecteurs supports, lesquels déterminent le vecteur de poids optimal $\vec{w_o}$ et le biais optimal associé, $b_o$. En minimisant la fonction de coût $\Phi$, l'équation de l'hyperplan optimal $H_{opt}$ est alors défini par :

$$\Phi = \frac{||\vec{w}||^2}{2} \qquad H_{opt} : \vec{w_o}\vec{x} + b_o \tag{3.9}$$

**Séparateurs linéaires - Cas non linéairement séparable :**

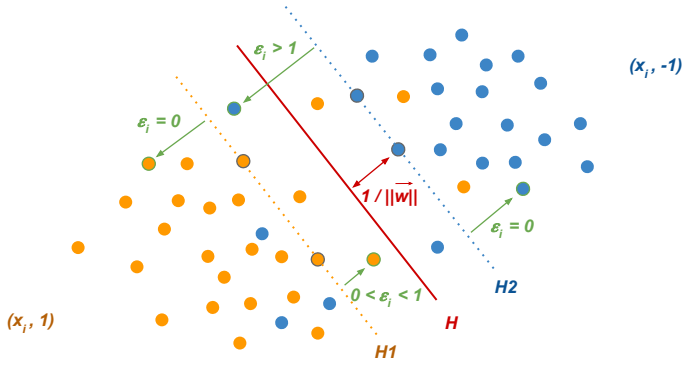



FIGURE 3.6 – Séparation de classes par les SVM dans le cas de données non linéairement séparables

Dans des cas où, tel qu'illustré par la figure 3.4b, les données ne sont pas linéairement séparables dû à du bruit comme dans la plupart des problèmes réels, la notion de marge souple est introduite. Ainsi, aux formules précédentes s'ajoute un ensemble de variables d'ajustement, "$\epsilon_i$" (marge d'erreur), contrôlées par un paramètre de régularisation, "$C$", lequel étant une constante pondérant cette erreur. La formulation de l'équation 3.6 d'un hyperplan, et la fonction "$\Phi$" à minimiser pour trouver l'hyperplan optimal deviennent :

$$\begin{cases} Minimiser \;\; \Phi = \dfrac{||\vec{w}||^2}{2} + C \sum_{i}^{l} \epsilon_i \\ Tel\ que \;\; f(x_i) = y_i(\vec{w}.\vec{x_i} + b) \geq 1 - \epsilon_i \end{cases} \tag{3.10}$$

Le paramètre $\epsilon_i$ défini la localisation de la donnée "i" par rapport à la marge, tel que le montre la figure 3.6. Une valeur à zéro établi que l'observation est du bon côté de la marge ou sur la marge, une valeur supérieure à zéro indique que l'observation est du mauvais côté et une valeur supérieure à un défini une donnée du mauvais côté de l'hyperplan. Le paramètre $C$ contrôle le compromis entre la largeur de marge et les erreurs de classification. Par conséquent, plus $C$ sera petit, moins les erreurs seront prises en compte et plus la marge sera large. Plus $C$ sera grand, plus les erreurs de classification seront considérées comme importante et plus la marge sera restreinte.

**Séparateurs non linéaires :**
Dans la majorité des cas en application réelle, tel qu'illustré par la figure 3.4c, le problème de répartition de données est plus complexe et la frontière ne peut être linéaire. Pour répondre à une telle problématique, une fonction noyau non linéaire, "$K(x, y)$", est introduite, permettant de représenter les données dans un nouvel espace où il est possible de les séparer par un hyperplan, tel que l'illustre la figure 3.7. Cette fonction noyau "$K$" est appliquée aux données et est utilisée dans le problème d'optimisation quadratique pour déterminer les nouveaux vecteurs supports. La fonction noyau $K(x, y)$ peut prendre différentes formes, et le choix de cette forme est généralement effectué de manière empirique selon l'application. Chacune de ces formes introduit de nouveau paramètres, ($\gamma$, $\theta$, $q$), qui sont eux aussi à optimiser lors de l'apprentissage.

Les trois principaux noyaux utilisés dans la littérature sont :

— **Fonction de base radiale gaussienne *(gaussian Radial Basis Function - RBF)* :**

$$K(\vec{x}, \vec{y}) = e^{(\gamma\|\vec{x}-\vec{y}\|^2)} \tag{3.11}$$

— **Noyau Polynomial (d'ordre "$q$") :**

$$K(\vec{x}, \vec{y}) = [\gamma(\vec{x}.\vec{y}) + \theta]^q \tag{3.12}$$

— **Noyau Sigmoïde ou tangente hyperbolique :**

$$K(\vec{x}, \vec{y}) = tanh[\gamma(\vec{x}.\vec{y}) - \theta] \tag{3.13}$$

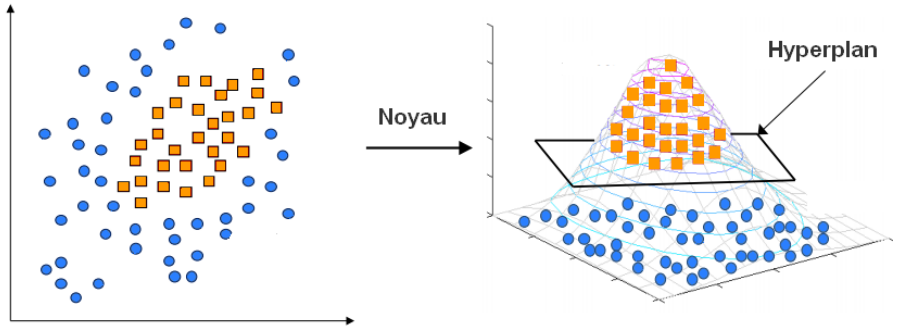



FIGURE 3.7 – Noyau appliqué aux SVM dans le cas de données non linéairement séparables[1]

### Forêts Aléatoires *(RF - Random Forest)*

Comme défini précédemment, un arbre de décision est une structure où chaque attribut du jeu de données représente un "nœud interne", chaque "branche" représente le résultat d'un test sur la valeur de cet attribut, et les "feuilles", présentes en bout de branches, sont issues de l'enchainement des diverses décisions prises en fonction des nœud d'une branche. L'algorithme des forêts aléatoires *(Random Forest - RF)* [108, 109] est un algorithme d'apprentissage supervisé capable d'effectuer des tâches soit de classification, soit de régression. Comme son nom l'indique, cet algorithme représente une "forêt", ou un ensemble d'arbres de décisions uniques, comme le montre la figure 3.8. Cet algorithme fait appel au *"bagging" (ou Bootstrap Aggregating)*, l'idée générale reposant donc sur le fait qu'une combinaison de modèles d'apprentissage améliore le résultat global par rapport à un unique modèle. Ainsi, chaque classifieur de base (ici un arbre de décision) est entraîné à partir d'un unique jeu de données issu des données d'apprentissage.

Le *"Bootstrapping"* est donc une technique d'échantillonnage permettant de construire des sous-ensembles de données en sélectionnant aléatoirement, avec remise, des données dans l'ensemble principal contenant les données d'apprentissage. Par conséquent, à partir d'un ensemble "$E$" de "$K$" caractéristiques, sont créés des sous-ensembles de "$m$" caractéristiques, aléatoirement sélectionnées. Le plus souvent on choisit "$m$" de telle sorte que $m = \sqrt{K}$. Par la suite, $n$ arbres de décisions sont alors créés à partir de $n$ sous-ensembles, construits de la manière précédente, chacun représentant alors un ensemble de données d'apprentissage pour l'arbre de décision associé.

1. Image source : hackerearth.com - "simple tutorial svm parameter tuning"

Chaque nœud de l'arbre, donnant lieu à deux branches, est alors défini afin de séparer au mieux les données au travers de ces branches, et donc de minimiser l'entropie (maximisant alors le gain) que nous définirons par la suite. La prédiction finale est donc issue d'un vote à la majorité (moyenne) des décisions des $X$ arbres ($P_i(y|E_i)$, $\forall i \in [1;n]$). Ces derniers *(weak learner)* introduisent une haute variance conduisant souvent au surapprentissage, et le procédé cherche à augmenter la dissimilarité entre les arbres ce qui permet de moyenner cette variance et de construire un classifieur fort *(strong learner)*.

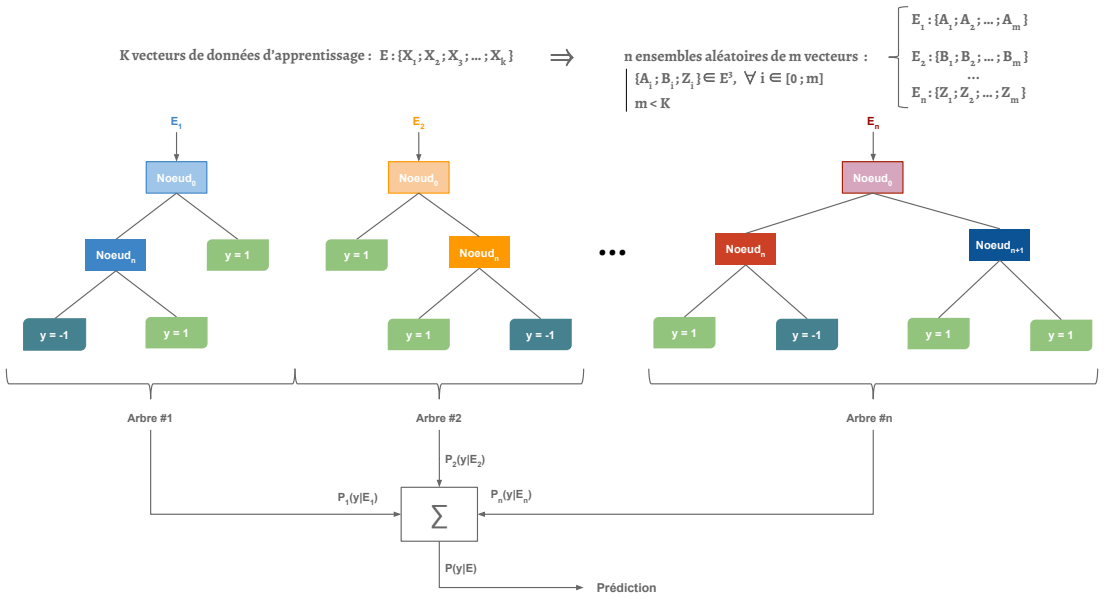



FIGURE 3.8 – Schématisation du principe des Random Forest pour une classification binaire

Le nœud racine de l'arbre $n$ dispose de tous les vecteurs de l'ensemble $E_n$ qu'il va alors séparer en deux parties à l'aide d'une certaine condition sur un élément de ces vecteurs. Les nœuds suivant, disposant chacun d'un de ces nouveaux sous-ensembles, réitèrent le même procédé avec une nouvelle condition sur un nouvel élément des vecteurs de ces sous-ensembles, jusqu'à atteindre une limite définissant une "feuille". Cette division des ensembles de données se doit d'être optimale (chaque nouveau sous-ensemble doit maximiser les éléments d'une classe par rapport à l'autre) afin de converger au plus vite vers une décision ayant une forte probabilité. Par conséquent, le critère et l'élément des vecteurs sur lequel imposer ce critère doivent être choisis avec précaution, à l'aide d'une "fonction d'impureté" $f_{imp}(j)$". Cette fonction, à minimiser pour scinder un nœud, mesure la "pureté" de l'ensemble de données découlant d'un nœud et peut s'écrire de deux manières différentes. La première, $i_H(j)$, basée sur l'entropie de Shannon, et la seconde, $i_G(j)$, basée sur l'indice de Gini, sont définies par l'équation 3.14, où "$y$" est le nombre de classes et "$p(y_i|j)$" est la fréquence ou la probabilité d'apparition de l'étiquette "$y_i$" au nœud "$j$". Pour définir la condition de séparation optimale, une fonction de gain, $G(j)$, est calculée à partir de la fonction d'impureté du nœud actuel et de celles des nœuds suivants. Cette fonction $G$, qui doit être maximale pour scinder un nœud, vérifie que les nœuds "enfants" représentent une source d'informations plus importante que leur parent.

Dans son équation, "$j$" représente l'indice du nœud parent, "$l$" l'indice du nœud enfant et "$E_j$" le nombre d'éléments de l'ensemble de données au nœud "$j$" :

$$f_{imp}(j) = \begin{cases} i_H(j) & = \sum_{i=1}^{y} -p(y_i|j).log_2(p(y_i|j)) \\ i_G(j) & = \sum_{i=1}^{y} p(y_i|j)(1-p(y_i|j)) \end{cases} \quad \Rightarrow \quad G(j) = f_{imp}(j) - \sum_{k=j}^{l} \left( \frac{E_k}{E_j} . f_{imp}(k) \right) \tag{3.14}$$

Après l'apprentissage, pour la classification d'un nouvel échantillon "$x$", la décision finale des RF peut être donnée soit par une moyenne, $< P(x,y|E) >$, sur les $n$ prédictions $P_i(x,y|E_i)$ des $n$ arbres, soit par un vote à la majorité, $\hat{P}(x,y|E)$, selon ces mêmes prédictions :

$$< P(x,y|E) >= \frac{1}{n} \sum_{i=1}^{n} P_i(x,y|E_i) \quad \Big| \quad \hat{P}(x,y|E) = \max_{j=1}^{y} \sum_{i=1}^{n} P_i(x,y|E_i) \tag{3.15}$$

Lors de l'apprentissage des RF, les paramètres de la table 3.2 permettent d'initialiser les arbres de décisions ainsi que de limiter la croissance des arbres et stopper l'apprentissage (critères d'arrêt).

| | |
|---|---|
| **Nombres de classes** | Nombre de classes dans l'ensemble d'origine "$E$" |
| **Nombre d'échantillon** | Nombre de vecteurs présents dans l'ensemble d'origine "$E$" |
| **Nombre d'éléments** | Dimension d'un vecteur de données |
| **Nombre de données aléatoires** | Quantité de vecteurs sélectionnées aléatoirement parmi "$E$" |
| **Profondeur maximale** | "Étages" maximum par arbre (critère d'arrêt d'un arbre) |
| **Échantillons minimum** | Définit un nœud comme une feuille si le nombre de vecteurs restant est inférieur à cette valeur (critère d'arrêt d'un arbre) |
| **Nombre d'arbres** | Nombre maximum d'arbres (critère d'arrêt des RF) |
| ***out-of-bag error* minimal** | Erreur estimée sur l'ensemble des RF (critère d'arrêt des RF) |

TABLE 3.2 – Paramètres d'apprentissage des Forêts Aléatoires

## 3.3/ Deep Learning

Comme précisé au début de ce chapitre, le *Deep Learning*, ou apprentissage profond, (également connu sous le nom de *deep structured learning*, *hierarchical learning* ou *deep Machine Learning*) est une branche du *Machine Learning*. Ce sous-ensemble du *deep Machine Learning* repose sur le principe des réseaux de neurones artificiels (*Artificial Neural Network - ANN*), utilisés cependant à une échelle beaucoup plus importante, reposant sur le fait qu'une augmentation du nombre de couches et de neurones d'un ANN augmente les performances de classification. Théorisé à l'origine en 1986 [110, 111], le concept du *Deep Learning* n'est devenu populaire que récemment [112], exigeant un très grand nombre de données annotées et une grande puissance de calculs.

***Artificial Neural Network - ANN* :**
Un réseau de neurones artificiels est un réseau composé de neurones artificiels interconnectés, où chaque neurone représente une unité de traitement d'information, comme le montre la figure 3.9.

Ces neurones traitent et propagent l'information aux autres d'une manière similaire aux neurones du cerveau humain. Chacun reçoit un ensemble d'informations auxquelles il applique une certaine opération avant de transmettre le résultat aux neurones suivant. Les neurones sont rassemblés en couches formant des niveaux hiérarchiques. Les neurones recevant l'information "brute" ou originale font partie de la couche appelée couche d'entrée et, de la même façon, les neurones fournissant les données de sortie du réseau font partie de la couche appelée couche de sortie. Tous les autres neurones, aux travers desquels l'information se propage de l'entrée à la sortie, font partie des couches cachées.
Sur chaque lien entre les neurones, l'information transmise est pondérée. Ainsi, un neurone reçoit un certain nombre de données de la couche précédente et utilise une fonction de transfert, sommant ces données pondérées, avant de traiter l'information. Le résultat est ensuite transmis à une fonction non linéaire, appelée "fonction d'activation", laquelle utilise une constante de seuil (aussi appelée "biais") afin d'éviter de transmettre une information nulle aux neurones de la couche suivante. Lors de l'apprentissage, les poids sont ajustés en fonction des performances du réseau.

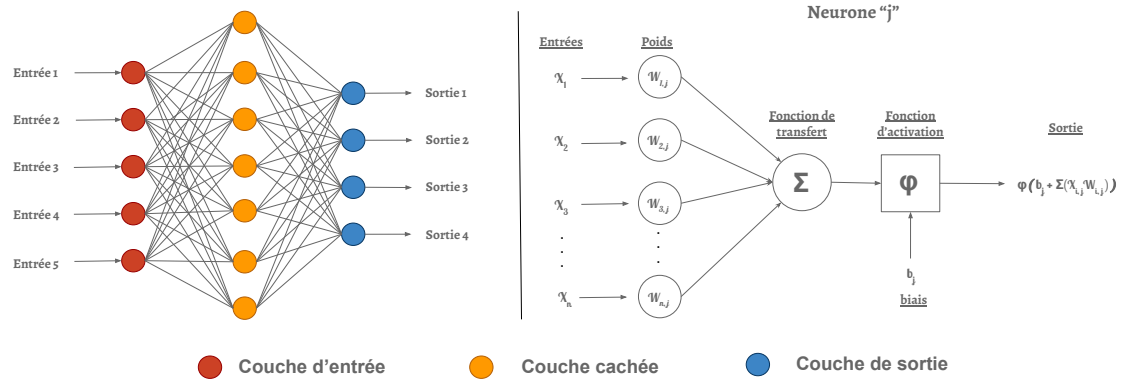



FIGURE 3.9 – Principe d'un réseau de neurones artificiels

On peut catégoriser les réseaux de neurones les plus simples selon deux familles : réseaux *"Feedforward"* et réseaux *"Feedback"*. Pour le type *"Feedforward"*, l'entrée est directement propagée vers la sortie (dans une seule direction). Pour le type *"Feedforward"* (ou récurrents), la sortie de certains neurones est retournée vers l'entrée de neurones précédents ("boucle") permettant ainsi d'augmenter les performances.

**Réseaux de neurones profonds *(Deep Neural Network - DNN)* :**
Les réseaux de neurones, comme présenté précédemment, comportent un nombre limité de neurones et disposent par conséquent de capacités limitées face aux problèmes de la littérature de plus en plus complexes. Afin de répondre à ces problèmes, les structures du Machine Learning doivent évoluer afin de proposer des modèles d'apprentissage plus complexes, capables de traiter des informations fournies par des milliers voir des millions de données. Une solution consiste à concevoir des réseaux de neurones disposant d'un plus grand nombre de couches cachées (ainsi qu'un plus grand nombre de neurones par couche), comme le montre la figure 3.10, que l'on appelle alors des réseaux de neurones profond [113, 114]. Cette augmentation des capacités et de la complexité du réseau augmente par conséquent la quantité de poids des neurones et la quantité d'informations propagées, conduisant nécessairement à une augmentation du nombre de calculs et donc des besoins en ressources informatiques.

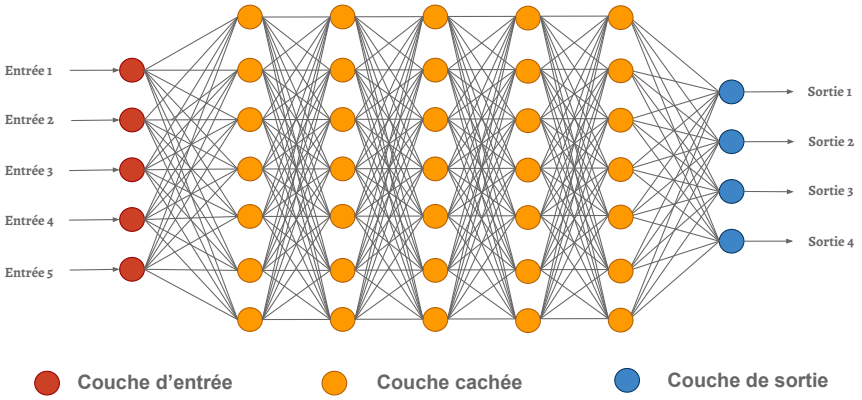



FIGURE 3.10 – Deep Learning Neural Network

Il existe de nombreux types de réseaux de neurones et réseaux profonds suivant leur architecture. Les plus populaires dans la conception des réseaux de neurones profond étant :

— **Perceptrons multicouches :**
Ce sont les réseaux les plus basiques parmi les réseaux "*feedforward*". Ils utilisent généralement des fonctions d'activations non linéaires. Ils peuvent être utilisés comme une partie d'un plus grand réseau de neurones profond.
— **Réseaux de neurones récurrents *(Recurrent Neural Networks - RNN)* :**
Les RNN sont aussi catégorisés comme des réseaux "*feedforward*", cependant ils possèdent des boucles de connexions récurrentes, propageant le résultat d'un neurone au précédent ou à lui-même. Ainsi, le réseau conserve en "mémoire" tout ou partie des informations précédentes et est donc capable de les utiliser afin d'affiner les résultats suivants. Ces réseaux sont surtout utilisés à des fins prédictives comme reconnaissance de texte ou en traduction.
— **Réseaux de neurones convolutifs *(Convoluted Neural Network - CNN)* :**
Les CNN sont similaires aux réseaux de neurones profonds ordinaires mais leur architecture est spécifique au traitement d'images. En effet, l'information de l'image est traitée en différents points par une convolution avec plusieurs filtres pour chaque couche. La récente popularité du *Deep Learning* est due à ces réseaux, permettant entre autres les dernières avancées en matière de conduite autonome, d'analyse complexe d'images, etc.

### 3.3.1/ RÉSEAUX NEURONAUX DE CONVOLUTIONS (CNN)

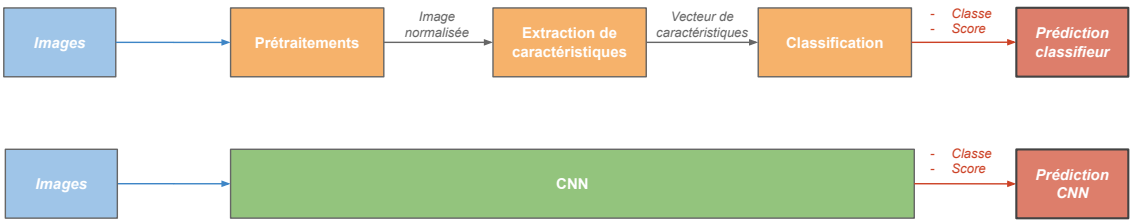



FIGURE 3.11 – Différence entre CNN et Machine Learning Traditionnel pour l'analyse d'images

Dédiés à l'analyse d'images, les réseaux neuronaux de convolutions (CNN) embarquent toute la chaîne de traitements décrite par la figure 3.2. Contrairement au Machine Learning classique et comme présenté par la figure 3.11, ces CNN peuvent être considérés comme une boite noire utilisant un ensemble d'images d'apprentissage de même dimensions pour ajuster les nombreux paramètres du réseau et ainsi se spécialiser à une certaine tâche. Nous nous proposons ici de présenter rapidement les éléments constituant un CNN. Afin de faciliter la compréhension du fonctionnement de ces CNN, l'annexe A.2.1 détaille les éléments présents dans cette "boite noire" et les différents concepts associés.

La figure 3.12 présente une schématisation de l'interaction entre les diverses couches du réseau avec une image d'entrée en couleur sur trois canaux. Un CNN est divisé en plusieurs couches, chacune composée d'autres couches représentant des tâches fondamentales du réseau. La partie réalisant l'extraction des caractéristiques d'une image est composées de couches appelées "couches cachées", et la partie classification est appelée couche de classification ou couche dense.
Une couche cachée comporte une ou plusieurs couches de convolutions, associées à une fonction d'activation, et une couche de sous échantillonnage (*"pooling"*). La fonction d'activation non linéaire, de la même manière que pour les simples ANN, permet de rectifier les valeurs des données par une normalisation. De nombreuses fonctions existent, les plus utilisées dans la littérature étant la fonction ReLU (*Rectified Linear Unit*), la fonction Sigmoïde ou la fonction tangente hyperbolique. La couche de sous échantillonnage, comme son nom l'indique, permet de réduire la quantité de données en sortie de la couche de convolution. Différents types sous échantillonnage existent, tels que le moyennage local ou le maximum local.
La couche de classification se divise en deux couches. Une première, la couche de vectorisation combine les caractéristiques locales détectées par les couches précédentes, s'affranchissant alors de leur structure spatiale. Une seconde couche, la couche entièrement connectée, permet une classification des caractéristiques générées par le réseau, qui ont alors été transformées en un vecteur de données. Une dernière fonction d'activation, souvent de type *"SoftMax"*, permet de normaliser les scores associés à chaque classe.

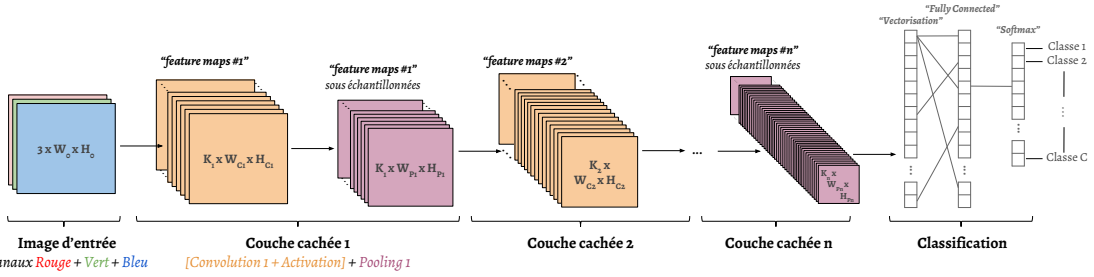
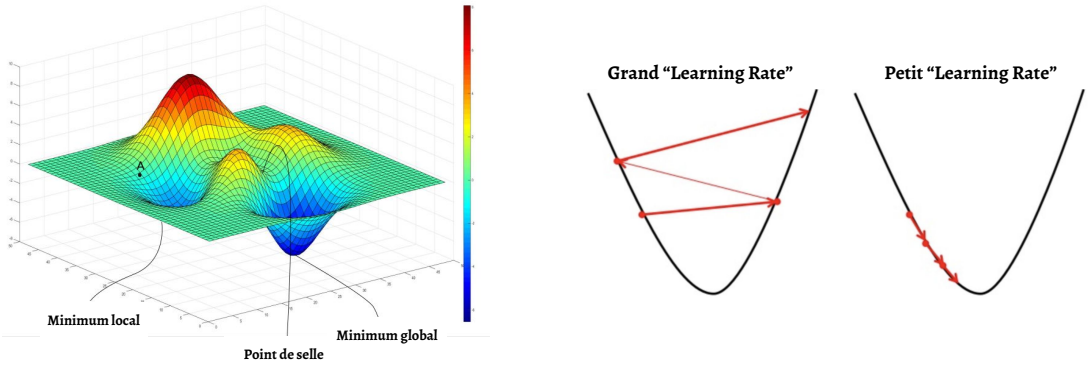



FIGURE 3.12 – Fonctionnement des couches d'un réseau CNN

Chaque couche cachée "$i$" est constituée de $K_i$ filtres de convolutions (dont les noyaux sont de même taille pour une couche), produisant alors $K_i$ nouvelles images (appelées *"feature maps"*) de dimension $W_{Ci}.H_{Ci}$ inférieure à la dimension de l'image d'entrée $W_{i-1}.H_{i-1}$. Une fonction d'activation s'applique ensuite à ces *"feature maps"*, qui sont par la suite sous échantillonnées, produisant $K_i$ images sous échantillonnées, de dimensions $W_{Pi}.H_{Pi}$. Un certain nombre de couches cachées s'enchainent jusqu'à la couche de classification. Les informations extraites avant classification, d'abord abstraites (contours), forment des caractéristiques haut niveau (représentant mieux l'image) au fur et à mesure de la progression de l'information au travers des couches.
Un réseau comporte différents paramètres définissant son architecture et leur nombre détermine la taille de stockage du réseau. Ces paramètres représentent les poids des noyaux de chaque convolution et de la couche de classification, ainsi que les biais associés. Lors de l'apprentissage, il

est possible de modifier des hyperparamètres, déterminant quant à eux le comportement du réseau sans modifier son architecture (tels que des critères d'arrêts ou la précision de l'ajustement des paramètres). Pour une meilleure compréhension des interactions entre les couches, C. Olah [115, 116] propose une visualisation poussée de la progression (ou propagation) d'une image au travers du réseau GoogLeNet.

Lors de l'apprentissage, les paramètres du réseau sont initialisés aléatoirement et les images d'apprentissage sont propagées dans le réseau. Avec les étiquettes de données associées, les performances de classification sont évaluées par le calcul d'une fonction de perte (ou fonction de coût). Cette fonction mesure l'erreur de classification et donc l'écart entre la probabilité de la prédiction et la vérité de terrain. Les poids et les biais du réseau sont ensuite ajustés par un algorithme d'optimisation itératif, appelé descente de gradient, de telle sorte qu'une nouvelle évaluation produise un déplacement de l'erreur vers un minimum local ou global du gradient de cette fonction de coût (figure 3.13a). Le pas de ce déplacement à chaque itération est appelé *"learning rate"*. Ce pas est un hyperparamètre pouvant être réglé afin d'influencer la convergence de l'apprentissage. Tel que le décrit la figure 3.13b, un grand pas permet de couvrir une plus grande région du gradient mais des pentes vers un minimum, pouvant être très brèves, peuvent être manquées. Un petit learning rate est plus précis et permet de capter plus facilement les variations de pente, mais induit un coût important en temps de calcul, fortement lié à la quantité de paramètres à ajuster. Un compromis est donc indispensable.




(a) Illustration 3D de la descente de gradient (b) Effet du *Learning Rate*

FIGURE 3.13 – Optimisation par descente de gradient

Il existe plusieurs types de descente de gradient, se différenciant principalement dans la quantité de données qu'ils utilisent (appelée *"batch"*). Les trois principaux étant les suivants :

- **le *batch gradient descent* :** l'erreur est calculée pour chaque images de l'ensemble d'apprentissage (*m* images) et le réseau n'est optimisé qu'après (batch=*m*),
- **la descente de gradient stochastique :** Les paramètres sont ajustés pour chaque image d'apprentissage (batch=1),
- **le *mini batch gradient descent* :** les données d'apprentissage sont divisées en groupes de *n* images, et l'optimisation intervient après le calcul de l'erreur de ces images (batch = n).

La taille du *mini batch* pour le *mini batch gradient descent* peut être ajustée en tant que paramètre d'apprentissage du réseau, au même titre que le *Learning Rate*. La quantité de fois où la totalité des images d'apprentissage sont présentées au réseau correspond au nombre d'étapes d'apprentissage (ou *"d'epoch"*), et le nombre d'itérations correspond au nombre de batch nécessaires pour compléter une epoch.

### 3.3.2/ Transfert Learning

En général, peu de personnes entraînent un CNN entièrement (apprentissage à partir d'un réseau vierge, c'est à dire un réseau dont les poids n'ont pas été définis) du fait de la difficulté d'obtenir le grand nombre d'images annotées nécessaire à ce type de réseaux. Il est plus commun de trouver une exploitation des CNN à partir du *Transfer Learning*. Le *Transfer Learning* est une technique d'apprentissage, surtout utilisée en Deep Learning, permettant d'entraîner un réseau à une certaine tâche à partir d'un modèle déjà entraîné sur une tâche similaire (e.g réseau original, $m$ classes : classification d'animaux ; nouveau réseau $n$ classes : classification de races de chats ; avec $n$ pouvant être différent de $m$). Les connaissances de ce réseau pré-entraîné sont transférées au nouveau problème afin d'assister l'apprentissage à partir de nouvelles images, et certaines étapes du réseau sont alors affinées (*"Fine-Tuning"*). Cette forme d'apprentissage présente l'avantage de nécessiter moins de données (milliers au lieu de millions) et permet un apprentissage beaucoup plus rapide, passant de centaines ou milliers d'heures de calculs à quelques heures voire quelques minutes. Comme l'apprentissage d'un CNN peut prendre plusieurs semaines, même avec la puissance de calcul de plusieurs GPU (*Graphics Processing Unit*), il est courant que les grandes structures de recherche rendent disponibles certains de leurs réseaux sur le net. Ces réseaux sont majoritairement entraînés sur les images du challenge ImageNet avec 1000 classes en reconnaissance d'objets.

Un CNN pré-entraîné peut être exploité de différentes façons par le Transfer Learning en fonction de la taille du nouveau jeu de données d'entrée et de la similarité des nouvelles images avec celles utilisées lors de l'apprentissage original. Les trois principales formes d'apprentissage par transfert de connaissances sont les suivantes :

- ***Fine-tuning* totale :** Dans ce cas de figure, la dernière couche entièrement connectée (couche de classification) est remplacée par un classifieur adapté au nouveau problème. Toutes les couches du réseau sont ensuite ré-entraînées sur les nouvelles images. Cette stratégie est utilisée lorsque la nouvelle collection d'images est grande. Comme les poids sont initialisés avec les valeurs du réseau pré-entrainé puis affinés, ce nouvel apprentissage est plus rapide.
- ***Fine-tuning* partiel :** De la même manière que précédemment, la dernière couche entièrement connectée est remplacée par un nouveau classifieur mais seules certaines couches de l'extraction de caractéristiques sont ré-entraînées, les poids des premières couches étant maintenues à leurs valeurs. Les premières couches d'un réseau étant en effet capables d'extraire des caractéristiques abstraites, ou génériques (contours, couleurs, etc.) de l'ancien jeu d'images d'apprentissage, et les dernières couches produisant progressivement des caractéristiques plus spécifiques aux classes des données originales, ce sont les poids de ces dernières qui sont ajustés. Cette stratégie est utilisée lorsque la nouvelle collection d'images est plus petite et différente.
- **Extraction de caractéristiques :** Le dernier cas de figure consiste à se servir des couches extrayant les caractéristiques du réseau pré-entraîné afin de représenter les nouvelles images d'un nouveau problème. Ainsi, la dernière couche entièrement connectée est réinitialisée ou retirée pour être remplacée par un nouveau classifieur, et les paramètres des autres couches sont fixés. Le nouveau classifieur va alors être entraîné à partir des vecteurs de caractéristiques extraites par les autres couches du réseau pré-entraîné. Cette stratégie est utilisée lorsque la nouvelle collection d'images est petite et présente des similitudes aux images originales.

## 3.4/ Évaluation des méthodes

### 3.4.1/ Métriques d'évaluation

Les sections précédentes ont montré qu'il existe un grand nombre de solutions aux problèmes de classification d'images dans le domaine de la biométrie. Les méthodes présentées peuvent être utilisées pour des problèmes de classification à deux ou plusieurs classes, et le comportement des classifieurs dépend du nombre d'échantillons par classe ainsi que de leur composition. Par conséquent, la sélection de la méthode la plus appropriée est dépendante des contraintes liées à l'application ciblée. Une solution consiste à effectuer une première sélection de méthodes, à les tester puis à réaliser une série d'évaluations. Les techniques d'analyse des données utilisent habituellement comme point de départ une représentation sous forme de tableau, que l'on appelle matrice de confusion ou tableau de contingence, mettant en relation le nombre de prédictions "$X_{i,j}$" ($X_{classe,prediction}$) des échantillons de la classe "i" attribués à une classe "j" (parmi $C$ classes). Le nombre d'échantillons formant la classe "i" est noté "$K_i$", et le nombre total de prédictions attribuées à cette classe est nommé "$M_i$". La somme des "$K_i$" ainsi que la somme des "$M_i$" correspondant au nombre total d'échantillons ($\Sigma$).

| | | **Prédiction** | | | **Total / classes** |
|---|---|---|---|---|---|
| | | **Classe$_1$** | **Classe$_i$** | **Classe$_C$** | |
| **Classe réelle** | **Classe$_1$** | $X_{1,1}$ | $X_{1,i}$ | $X_{1,C}$ | $K_1$ |
| | **Classe$_i$** | $X_{i,1}$ | $X_{i,i}$ | $X_{i,C}$ | $K_i$ |
| | **Classe$_C$** | $X_{C,1}$ | $X_{C,i}$ | $X_{C,C}$ | $K_C$ |
| **Total prédictions** | | $M_1$ | $M_i$ | $M_C$ | $\Sigma$ |

Table 3.3 – Matrice de confusion des prédictions d'un classifieur à C classes

En considérant le problème précédent, pour chacune des classes "i", comme étant binaire (Classe "i" : positif ; Toutes les autres classes "j ≠ i" : négatif), ou directement dans le cas d'un problème à deux classes, les prédictions sont résumées sous la forme de quatre informations principales :

- **Vrais Positifs - VP *("True Positive" - TP)* :**
  Échantillons de la classe positive ("i") correctement classés $(X_{i,i})$
- **Faux Négatifs - FN *("False Negative" - FN)* :**
  Échantillons de la classe positive ("i") incorrectement classés $(X_{i,j},\ \forall j \neq i)$
- **Vrais Négatifs - VN *("True Negative" - TN)* :**
  Échantillons des classes négatives ("j") correctement classés $(X_{j,t},\ \forall (j,t) \in [1,C] \neq i)$
- **Faux Positifs - FP *("False Positive" - FP)* :**
  Échantillons des classes négatives ("j") incorrectement classés $(X_{j,i},\ \forall j \neq i)$

Pour un problème à N classes, considéré comme binaire pour le test de chacune des classes, il y a autant de matrices de confusion qu'il y a de classes, chacune représentant les performances de la classification vis à vis de la classe "i". La matrice de confusion pour un problème à deux classes établit ainsi le lien entre le nombre d'échantillons total (P) de la classe positive, le nombre d'échantillons total (N) de la classe négative, les quatre informations précédentes et par conséquent le nombre total d'échantillons classés positivement (Ppos) et négativement (Pneg). Cette matrice de confusion, tableau 3.4, donne une indication sur les résultats de classification. Cependant, du fait que la quantité d'échantillons par classe peut varier entre les différentes classes et des différentes problématiques, son exploitation n'est pas toujours facile.

| | | Prédiction | | Total / classes |
|---|---|---|---|---|
| | | **Classe positive** | **Classe négative** | |
| **Classe réelle** | **Classe positive** | VP | FN | P |
| | **Classe négative** | FP | VN | N |
| **Total prédictions** | | Ppos | Pneg | Σ |

TABLE 3.4 – Matrice de confusion des prédictions d'un classifieur à deux classes

Des critères de performances dérivent de cette matrice de confusion et permettent d'évaluer rapidement les performances de classification. Ces nouvelles métriques représentent des indications normalisées de ces performances sous forme d'un pourcentage. Il est ainsi possible d'analyser et de comparer les performances de différents classifieurs. Ces mesures sont les suivantes :

— **Sensibilité ou Rappel *(Sensitivity ou Recall)* :**
Proportion des solutions pertinentes correctement identifiées (taux de vrais positifs). Mesure la capacité du système à donner toutes les solutions pertinentes.

- *Cas général (problème à C classes)* :

$$\left.\begin{aligned} Sensibilite_i &= \frac{X_{i,i}}{K_i} \\ Sensibilite_{Systeme} &= \frac{\sum_{i=1}^{C} Sensibilite_i}{C} \end{aligned}\right\} \Rightarrow \frac{\textit{Échantillons correctement attribués à la classe i}}{\textit{Échantillons de la classe i}} \tag{3.16}$$

- *Pour un problème à 2 classes* :

$$Sensibilite = \frac{VP}{VP + FN} = \frac{VP}{P} \tag{3.17}$$

— **Spécificité *(Specificity)* :**
Proportion de solutions négatives correctement identifiées (taux de vrais négatifs). Mesure la capacité du système à reconnaitre toutes les situations négatives. C'est une mesure complémentaire de la sensibilité.

- *Cas général (problème à C classes)* :

$$\left.\begin{aligned} Specificite_i &= \frac{\sum_{j=1}^{C}\sum_{t=1}^{C} X_{j,t}}{\sum_{j=1}^{C} K_j},\ \forall (j,t) \in [1,C] \neq i \\ Specificite_{Systeme} &= \frac{\sum_{i=1}^{C} Specificite_i}{C} \end{aligned}\right\} \Rightarrow \frac{\textit{Échantillons correctement attribués aux classes j≠i}}{\textit{Échantillons n'appartenant pas à la classe i}} \tag{3.18}$$

- *Pour un problème à 2 classes* :

$$Specificite = \frac{VN}{VN + FP} = \frac{VN}{N} \tag{3.19}$$

— **Précision *(Precision)* :**
Proportion de solutions correctes parmi les prédictions positives. Mesure la capacité du système à refuser les solutions non-pertinentes.

- *Cas général (problème à N classes)* :

$$\left.\begin{aligned} Precision_i &= \frac{X_{i,i}}{M_i} \\ Precision_{Systeme} &= \frac{\sum_{i=1}^{C} Precision_i}{C} \end{aligned}\right\} \Rightarrow \frac{\text{Échantillons correctement attribués à la classe i}}{\text{Échantillons attribués a la classe i}} \quad (3.20)$$

- *Pour un problème à 2 classes* :

$$Precision = \frac{VP}{VP + FP} = \frac{VP}{Ppos} \quad (3.21)$$

— **Robustesse, exactitude ou justesse *(Accuracy)* :**
Proportion des solutions correctes (positives et négatives) parmi toutes les prédictions. Mesure la capacité du système complet à donner toutes les solutions correctes.

$$Justesse = \frac{VP + VN}{VP + VN + FP + FN} = \frac{P}{\Sigma} \Rightarrow \frac{\text{Échantillons correctement attribués}}{\text{Totalité des échantillons}} \quad (3.22)$$

— **F$_\beta$-mesure *(F$_\beta$-measure) or F-Score* :**
Combine la précision et le rappel de par leur moyenne harmonique. Fourni une mesure favorisant soit le rappel soit la précision dans le cas d'une classification à partir de données non équilibrées.

$$F_\beta = \frac{(1 + \beta^2)(Precision.Rappel)}{\beta^2.Precision.Rappel} \quad (3.23)$$

La plupart du temps, ces termes sont utilisés sous leur forme anglo-saxonne. Cependant nous avons choisi d'utiliser leur équivalent français dans ce manuscrit. Seul le terme *"accuracy"* ne trouve pas de réelle de traduction précise en français, celle-ci étant très proche du mot "précision". Parmi les traductions équivalentes possibles, nous avons opté dans la suite de ce manuscrit pour le terme "justesse", qui représente au mieux cette mesure à notre sens.

L'évaluation d'un système complet décrit sa faculté à discriminer une classe par rapport à une autre. Dans le cas d'un système à plus de deux classes, cette évaluation représente une "moyenne" de ces métriques pour chaque classe. Cette moyenne peut être obtenue de deux manières différentes. On peut en effet distinguer la "micro-moyenne" de la "macro-moyenne" (calcul utilisé dans les descriptions précédentes). Chaque donnée (nombre de bonnes réponses et de mauvaises pour tout le système) est prise en compte dans le calcul de la micro-moyenne. Celle-ci calcule d'abord la somme des valeurs les plus fines des termes du numérateur et du dénominateur (présents dans la matrice de confusion) puis établit le critère d'évaluation à partir de ces sommes. Dans le cas du calcul de la macro-moyenne, chaque catégorie est prise en compte (nombre de bonnes réponses et de mauvaises par classe). Comme présentée, elle calcule les critères d'évaluations de chaque classe puis en effectue la moyenne. Cependant, si la quantité d'informations évaluées n'est pas équilibrée entre les classes, ces métriques peuvent conduire à une mauvaise interprétation du comportement du classifieur. Il est donc nécessaire de comprendre ce que représentent ces métriques, et la manière dont il faut les interpréter, afin de quantifier les performances d'un système.

### 3.4.2/ Méthodologie d'interprétation des résultats

Chacune de ces métriques présente un pourcentage décrivant une certaine aptitude du modèle. Plus cette valeur de pourcentage est haute, plus modèle est performant. La sensibilité et la spécificité ne tiennent compte que des échantillons provenant d'une même classe de test (classe positive pour la sensibilité et classe négative pour la spécificité). Ainsi, les variations du nombre d'images de test par classe n'ont aucune influence sur ces métriques. Cependant, ce n'est pas le cas pour la précision et la justesse. En effet, la précision tient compte des échantillons de test déterminés comme positifs pour les deux classes et la justesse est une évaluation "globale" du modèle, considérant tous les résultats de prédiction (toute la matrice de confusion). Par conséquent, si les données de test ne sont pas équilibrées, leur interprétation peut être erronée car elle ne reflète pas la réalité de la situation. L'interprétation des performances d'un classifieur ne peux donc se faire en exploitant les métriques proposées précédemment une à une, elles sont complémentaires.

On retrouve dans la littérature l'exploitation des couples sensibilité/spécificité et précision/rappel (ou sensibilité), figure 3.14. À partir de ce premier couple est tracé la courbe ROC (*Receiver Operating Characteristic*), permettant de résumer un certain compromis entre le taux de faux positifs et le taux de faux négatifs de par l'utilisation de différents seuils. Ces seuils, appliqués sur les scores de prédiction, permettent d'ajuster ces métriques en considérant une prédiction comme juste seulement si son score associé est supérieur à ce seuil. La forme de la courbe et l'aire sous la courbe à partir d'un certain point détermine les performances du modèle. La courbe PRC (*Precision-Recall Curve*) est tracée à partir de ce deuxième couple de métriques et résume les possibles compromis entre la précision et la sensibilité, à l'aide des valeurs correspondant aux résultats de prédictions seuillés. Cette courbe est interprétée de la même manière que précédemment. La courbe ROC est appropriée dans le cas de données de test équilibrées sur chaque classe, tandis que la courbe PRC est plus appropriée aux cas non équilibrés. La mesure $F_\beta$ fait également office d'une mesure de compromis entre ces deux dernières métriques lorsque les données sont déséquilibrées mais au travers de son seuil $\beta$.

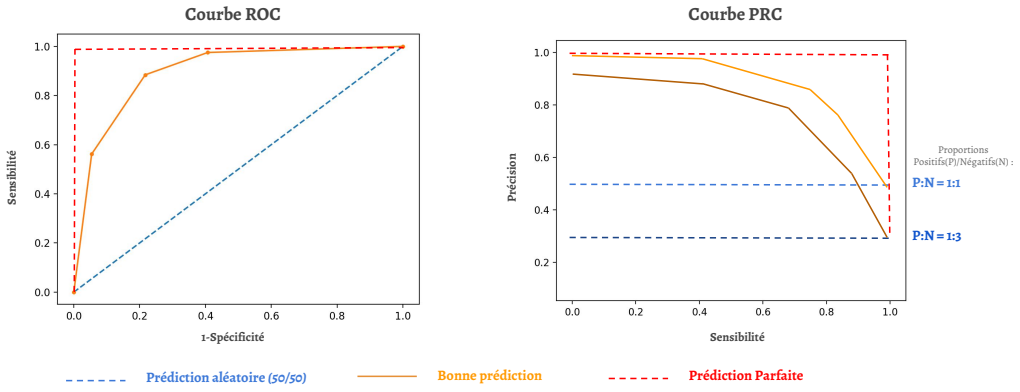



Figure 3.14 – Analyse des courbes ROC et PRC

Dans le cas d'un système à plus de deux classes, on peut exprimer ces métriques d'une manière différente en considérant un système de rang, que l'on pourrait comparer à "un podium". En effet, dans la littérature, et principalement pour challenge ImageNet mettant en œuvre une classification sur 1000 classes, les prédictions sont triées par ordre décroissant selon leur score et, si la classe de l'image testée se trouve parmi les cinq prédictions les plus probables, elle est considérée comme juste. Cela permet, en juxtaposant cette mesure à la mesure standard de la justesse, de relativiser les performances d'un classifieur au vu du grand nombre de classes.

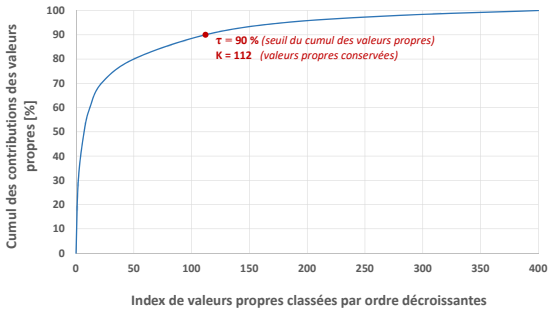
## 3.5/ Application à notre problématique

Le chapitre précédent a permis de justifier nos choix quant aux modalités biométriques sélectionnées pour répondre à la problématique de sécurisation d'accès. Avec une focalisation sur l'analyse d'images pour la biométrie, différentes méthodes du Machine Learning traditionnel et du Deep Learning ont jusque-là été exposées de manière générale. Nous allons désormais présenter leur utilisation dans le cadre des applications liées au projet de cette thèse. Ainsi, nous exposerons tout d'abord différentes optimisations à l'extraction de caractéristiques permettant de diminuer (ou réduire) la quantité des données et par conséquent la taille de stockage. Nous verrons ensuite, dans la littérature, les méthodes permettant de détecter les visages et comment sont effectuées leur authentification ou leur reconnaissance. Nous observerons ensuite différentes méthodes appliquées aux empreintes digitales et aux réseaux vasculaires, et nous effectuerons un choix sur les méthodes exploitées dans nos contributions.

### 3.5.1/ Réduction de données dans un contexte fortement contraint

Dans le domaine de l'analyse de données, et plus spécialement dans l'analyse d'image, la quantité de données informatives peut rapidement se révéler conséquente. En effet, pour une image, chaque pixel représente une certaine information, au même titre que les diverses relations entre ces pixels. De plus, toute l'image ne représente pas une information pertinente et du bruit peut également être présent. Par conséquent, comme présenté précédemment, une étape d'extraction de caractéristiques permet de déterminer ces relations décrivant les données. Cette extraction est donc suivie d'une sélection permettant d'isoler les caractéristiques séparant au mieux les classes d'un jeu de données. Ces étapes, nécessaires au processus de classification, réduisent la quantité de données en représentant l'information de manière optimale, mais ce n'est parfois pas suffisant.
Dans notre contexte, un des objectifs est d'obtenir des modèles de classification rapides (fonctionnement temps réel) et légers (taille de stockage inférieure à 8 Ko). Par conséquent, nous avons des besoins supplémentaires en matière de réduction de données (dimensions et nombre de caractéristiques, précisions de la représentation des paramètres du modèle, etc.), conduisant alors à une réduction de la complexité des calculs et une diminution des capacités de mémoire nécessaires au stockage d'un modèle. L'analyse en composantes principales (ACP) présentée précédemment, réalisant le processus d'extraction de caractéristiques, permet d'effectuer une réduction de dimension supplémentaire à la sélection de caractéristiques. En effet, de par son fonctionnement, cette méthode crée un espace propre représentant les données d'une manière optimale. Cet espace et composé d'un certain nombre de dimensions (vecteurs propres) dépendant directement de la quantité d'images utilisées pour créer cet espace. Une valeur propre est associée à chacun de ces vecteurs et représente en quelque sorte sa contribution dans l'espace par rapport aux autres. En triant ces valeurs propres dans l'ordre décroissant et en sommant chacune avec les précédentes, on obtient un "cumul des contributions des valeurs propres", observable figure 3.5 sous forme de pourcentage par rapport au cumul total. Turk et Pentland [117] proposent de réduire le nombre de valeurs propres conservées $K$ en appliquant un seuil $\tau$ sur ce cumul, en suivant l'équation 3.24 où $N$ est le nombre de valeurs propres total et $\lambda$ est une valeur propre.

L'impact des méthodes de réduction d'informations quant à l'adéquation algorithme et architecture, au sens stockage des données, est un point peu étudié dans la littérature. Cette remarque concerne majoritairement le Deep Learning pour lequel, du fait de son intérêt récent relativement au Machine Learning et de sa progression rapide, la littérature traite principalement de la théorie et des performances brutes. Cependant, au moment de réalisation de ces travaux, de nouvelles études commencent à s'intéresser à cet aspect [118, 119].




$$\sum_{i=1}^{K}(\lambda_i) \leqslant \tau . \sum_{i=1}^{N}(\lambda_i) \tag{3.24}$$

TABLE 3.5 – Cumul des contributions des valeurs propres de l'ACP - Exemple avec 400 valeurs propres au total ($N$) et un seuil ($\tau$) à 90% donnant 112 valeurs propres conservées ($K$)

En effet, avec la démocratisation de l'Internet des objets et leurs besoins grandissant en matière de prises de décisions et d'analyse de données, les méthodes rendant ces objets "intelligents" se doivent d'être optimisées pour de faibles puissances de calculs tout en maintenant des performances satisfaisantes [120, 121]. D'après certaines études [122, 123], les diverses optimisations de CNN pour répondre à ces nouvelles contraintes se résument au travers du *"pruning"* des réseaux lors de l'apprentissage, suivi de leur quantification et de leur optimisation. Le *"pruning"* [124, 125], lors de l'apprentissage d'un CNN, supprime les paramètres les moins pertinents à chaque itération de la rétro-propagation de l'erreur par la descente de gradient. Son but est de réduire la complexité du réseau et de prévenir du sur-apprentissage avec une meilleure généralisation. Partant de l'hypothèse que de nombreux paramètres sont redondants ou au minimum moins pertinents, cette technique supprime les connexions sans importance et permet donc de réduire la taille de stockage du réseau. La quantification [126, 127], intervenant après l'apprentissage, permet de "compresser" un réseau en limitant le nombre de bits représentant chaque poids, entraînant par conséquent une légère baisse de performances lors de la prédiction.

## 3.5.2/ EXTRACTION ET CLASSIFICATION DE DONNÉES BIOMÉTRIQUE

### 3.5.2.1/ MESURES DU VISAGE

**Détection de visages**

Avant de considérer une quelconque reconnaissance ou authentification de visages, il est nécessaire de détecter ces derniers. L'augmentation des performances des algorithmes de classification passe par la maximisation de la quantité de données propres aux visages par rapport à la quantité d'informations dans l'image traitée, et donc par la minimisation de l'influence des données extérieures au visage (arrière plan). La problématique de la détection de visages est étudiée dans la littérature depuis les premiers apports de la vision assistée par ordinateur. La recherche en matière de détection automatique des visages est en effet motivée par les diverses applications émergentes étroitement liées à ces besoins telles que la reconnaissance ou l'authentification de visages [128], le suivi de visages pour la surveillance [129], la reconnaissance des expressions faciales [130], la reconnaissance de genre et d'âge [131] ou encore la retouche automatique de photos du visage [132], etc. De plus, on retrouve la détection de visages embarquée dans la majorité des systèmes commerciaux disposant d'une caméra, utilisant cette technologie pour aider par exemple à la mise au point automatique, et également intégrée aux réseaux sociaux pour assister le marquage des personnes sur les photographies.

Cependant, cette problématique, appliquée à des images acquises en environnement non contraint, s'accompagne de nombreuses difficultés [133] (communes avec les différentes applications présentées précédemment) telles que : **l'orientation du visage** (variations d'orientations du visage par rapport à la position de face qui est la position optimale), **les expressions faciales**, **l'occultation totale ou partielle des visages** (d'autres personnes ou lunettes, barbe, écharpe, etc., apportant également aux visages des variations de formes et de couleurs), **les conditions environnementales de l'acquisition** (l'éclairage, s'il est mal positionné peut induire des ombres conséquentes sur le visage, générant alors de grandes variations par rapport à son aspect original) ou encore **l'échelle du visage sur l'image** (dû à des différences naturelles entre les humains, à des différences de positions et de distances par rapport à la caméra, ou à un objectif particulier tel qu'un *"fish eye"*).

Les travaux de recherche dans ce domaine ont connu une véritable explosion dans la décennie 1990 [134]. Cependant ces premières avancées sont inappropriées à une application réelle. Elles sont en effet trop peu robustes pour fournir des performances convenables sur des images de visages obtenues sans contraintes d'acquisition (également appelé *"in the wild"*). Cet objectif est atteint en 2001 avec les travaux précurseurs de Viola et Jones [135], décrivant une méthode basée sur du *"boosting"* [136], permettant alors pour la première fois d'approcher une détection fiable dans des conditions réelles. Cet algorithme a donné lieu à de nombreuses évolutions et déclinaisons, et est encore largement utilisé de nos jours. En 2002, Yang. M et al. [134] proposent de regrouper les différentes méthodes de la littérature pour la détection de visages en quatre catégories : *"Knowledge-based methods", "Feature invariant approaches", "Template matching methods"* et *"Appearance-based methods"*. Les méthodes *"Knowledge-based"* sont des méthodes supervisées utilisant des règles basées sur les connaissances humaines à propos de ce qui constitue un visage de manière générale. Les approches *"Feature invariant"* recherchent des structures du visage robustes aux variations d'éclairage et d'orientation du visage. Les méthodes *"Template matching"* utilisent plusieurs motifs généraux d'un visage, le décrivant dans son ensemble ou par seulement certaines caractéristiques, qui sont alors comparées avec l'image courante pour détecter le visage. Les méthodes *"Appearance-based"* utilisent des modèles appris à partir d'un ensemble d'images d'apprentissage, lesquelles représentent au mieux les diverses variations d'apparences d'un visage.

Bien que l'amélioration ultra rapide des performances des plateformes de calculs ait contribué à l'amélioration de la détection de visages, la progression des dernières années dans ce domaine est largement due à la communauté scientifique. Tout d'abord par l'introduction de méthodes robustes d'extraction de caractéristiques (SIFT [91], HoG [137], SURF [138], etc.) et au développement de nombreuses méthodes puissantes (boosting [139], SVM [100], réseaux de neurones [140], etc.). Mais également grâce aux nombreuses contributions en matière de bases de données et d'évaluation (LFW [141], FDDB [142], etc.), ainsi qu'au développement de bibliothèques et outils public mettant à disposition de nombreux codes et facilitant le développement de nouvelles méthodes (OpenCV [143], TensorFlow [144], etc.). La détection de visages peut désormais être considérée comme un procédé mature qui, selon Zafeiriou. S et al. [145] dans une étude de 2015, s'articule autour de trois grands axes de recherche principaux : les méthodes basées sur du *boosting (boosting-based methods)*, les applications des réseaux de neurones convolutionnels profonds (DCNN) et les méthodes *Deformable Parts-based Models (DPM)*. Les auteurs proposent de regrouper les méthodes récentes sous deux catégories majeures :

- **Algorithmes basés sur des modèles "rigides" incluant :**
  - des variations du *boosting* : principalement représentées par les travaux de Viola et Jones [135, 90] et leurs variations [133]
  - les algorithmes basés sur des CNN et des Deep CNN (DCNN) [146, 147, 148]
- **Algorithmes apprenant et appliquant un DPM** *(Deformable Parts-based Model)* [149, 150] afin de modéliser de potentielles déformations des parties du visage.

Les algorithmes basés sur un apprentissage de modèles "rigides" à l'aide du *boosting*, ainsi que les détecteurs basés sur des CNN, ont des difficultés à effectuer une détection dans le cas d'échantillons ne correspondant à aucune donnée déjà vue. Cependant, cette lacune est désormais largement atténuée par l'utilisation d'une très grande quantité d'informations, généralisant un visage en représentant chacune des difficultés énoncées précédemment et en utilisant plusieurs modèles "rigides" pour différentes orientations du visage. En comparaison, les méthodes basées sur les DPM modèlent les possibles déformations du visage et montrent donc une meilleure robustesse aux nouvelles vues. Une conséquence directe à ces méthodes est la limitation des données d'apprentissage, une généralisation complète du visage sous toutes ses orientations n'étant pas nécessaire.
L'étude de M. Mathias et al. [151] montre que les méthodes les plus performantes de chaque famille sont capables d'atteindre des résultats similaires. Dans leur travaux de 2014 combinant des idées provenant des deux approches [152], D.Chen et al. montrent que leur détecteur est à la hauteur des méthodes les plus performantes pour un temps de calcul de 33,8 millisecondes sur des images VGA en mono-cœur sur un processeur cadencé à 2,93 GHz. Cela représente un temps de calcul mille fois inférieur à l'état de l'art à cette date [153], approchant alors la vitesse du traitement de Viola-Jones (23 ms) qui présente alors des performances de détection largement inférieures mais qui demeure la méthode de référence encore aujourd'hui.

### Authentification/Reconnaissance de visages

Le classifieur ne permettant que de trouver une séparation optimale des données, les aptitudes d'une méthode de reconnaissance ou d'authentification de visages reposent essentiellement sur le type d'images utilisées (2D, 3D, thermique, etc.), ainsi que sur la méthode d'extraction de caractéristiques employée. Comme dans la section 3.2.1, on retrouve pour la reconnaissance/authentification de visages des méthodes de correspondances globales, locales et hybrides.

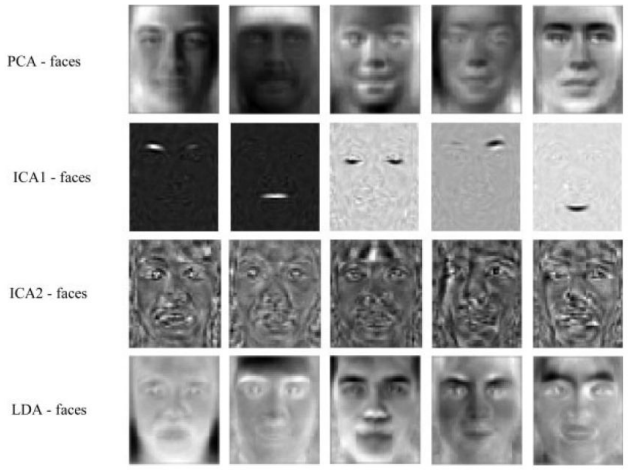


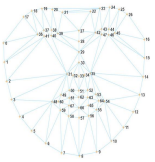

FIGURE 3.15 – Premières composantes, sous forme d'images, des espaces ACP, ACI et ADL [154]

Parmi les méthodes de correspondances globales du Machine Learning traditionnel, on retrouve l'analyse en composantes principales (ACP - *PCA*,) appliquée aux visages par Turk et Pentland en 1991 [117, 155] et nommée *"Eigenfaces"*. Leur implémentation extrait les caractéristiques de visages afin de construire une bibliothèque de vecteurs à laquelle est comparé, par une distance euclidienne, le vecteur de la projection d'une nouvelle image. Elle est souvent comparée à l'analyse discriminante linéaire (ADL - *LDA*), appliquée aux visages sous le nom des *"Fisherfaces"*, ainsi qu'à l'Analyse en Composantes Indépendantes (ACI - *ICA*) [156]. La figure 3.15 illustre la projection d'une image sur les premières composantes des différents espaces de visages, ACP, ADL et ACI ($ACI_1$ : caractéristiques locales et $ACI_2$ : caractéristiques globales).

Avec l'utilisation de différentes mesures de distances pour effectuer la correspondance entre les vecteurs d'apprentissage et les vecteurs de test, l'étude [154] démontrent que pour l'ACP et l'ADL, la mesure de distance $L_1$ produit les meilleurs résultats en terme de justesse. Elle est suivie de près par la distance $L_2$ (euclidienne) à moins de 2% de différence. Cependant, considérant l'espace des caractéristiques locales de l'ACI ($ACI_1$), c'est la distance de Mahalanobis qui produit la meilleure justesse. La distance de similarité cosinus fournit les meilleurs résultats pour l'espace de caractéristiques globales de l'ACI ($ACI_2$). L'étude [157] montre que les algorithmes de l'ACP et de l'ACI, respectivement associés à la mesure de distance $L_2$ et à la mesure de similarité cosinus, approchent les 100% de reconnaissance correcte sur leur bases de données de test.
L'ACP et l'ADL sont comparées par Shyam et al. [158], avec pour chacune, l'utilisation de la mesure de distance $L_2$ et la distance de Bray Curtis. Il est tout d'abord montré que ces deux méthodes d'extraction de caractéristiques ont des performances très proches. Mais la mesure de dissimilarité de Bray Curtis est légèrement plus efficace que la distance euclidienne sur certaines bases de données. Cependant elle fournit des résultats beaucoup plus stables sur des bases construites à partir d'exemples complexes, conservant plus de 90% de justesse.

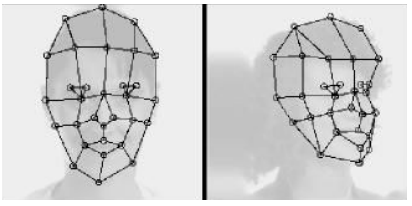
Dans le but de limiter les erreurs dues aux faiblesses de ces mesures de distances face à la possible complexe répartition des données, de nombreux classifieurs statistiques sont expérimentés, tels que les SVM, les Random Forest (RF), les réseaux de neurones ou les K-plus proches voisins (K-NN). L'étude [159] montre que, sur leur base de test, les SVM munis d'un noyau RBF appliqués à l'ACP et l'ADL fournissent des résultats plus stables et une classification plus juste qu'avec une quelconque mesure de distance. Leurs justesses sont respectivement proches de 100% avec un écart moyen de moins de 10% contre 80% avec un écart moyen de plus de 20%.

(a) *Landmarks*



(b) Triangulation de *landmarks*

(c) Histogrammes locaux de gradients orientés (HOG)



(d) *Elastic Bunch Graph Model(EBGM)*

FIGURE 3.16 – Illustrations d'extractions de caractéristiques locales

Les méthodes de correspondances locales utilisent des points caractéristiques (*"landmarks"*) tel qu'illustré par la figure 3.16a. Cette source d'informations peut être traitée de différentes manières comme avec une triangulation entre leurs coordonnées comme illustré par la figure 3.16b, en créant un modèle de liaisons "élastiques" par l'algorithme EBGM [160] figure 3.16d, ou encore en effectuant diverses mesures locales. Ces mesures sont souvent issues du calcul d'un histogramme local sur un découpage de la région d'intérêt du visage comme les LBP [161], les LTP [162] et leurs dérivés. Ce calcul d'histogramme peut être optimisé, comme le montre la figure 3.16c, en ne considérant que les régions contenues autour des landmarks [163] où l'histogramme des gradients orientés *("HOG")* est mesuré dans les régions des yeux, du nez et de la bouche.

Toutes ces méthodes sont régulièrement évaluées et comparées à l'état de l'art. Dans une étude de 2016 [78], un grand nombre de méthodes du Machine Learning traditionnel sont comparées à partir de diverses bases de données de visages dédiées à la recherche. Il ressort qu'à l'heure actuelle, la majorité des méthodes dépassent une justesse de 90% en matière de reconnaissance de visage, les méthodes de correspondances locales et les méthodes hybrides atteignant 100% de bonne classification sur des images simples avec peu de perturbations (telle que la base ORL [164]), et 95% sur des images plus complexes telle que la base FERET [165].

Pour finir, les CNN présentent désormais un apport conséquent dans l'état de l'art de la reconnaissance et de l'authentification de visages [166, 167]. De plus, un réseau peut être utilisé comme seul extracteur de caractéristiques ou comme chaîne complète de l'application (extraction et classification). Ces CNN utilisent des millions d'images d'apprentissage ce qui permet d'obtenir une généralisation complexe et précise des visages. En effet, contrairement au Machine Learning classique, plus le nombre d'images couvrant les différentes représentation d'une classe est important, plus les CNN seront aptes à trouver des caractéristiques optimales. Grâce à internet, l'apport massif d'images dans le processus d'apprentissage s'est vu facilité et a permis d'améliorer très rapidement les performances jusqu'à atteindre cependant une certaine saturation [168]. Chaque nouveau réseau montre en effet un gain de moins de 0,1% de justesse. Ces réseaux sont évalués dans l'état de l'art sur une base d'images de recherche complexe (*Labeled Faces in the Wild* - "LFW" [169, 170]), comportant un grand nombre d'images de visages acquises sans contraintes, principalement issues de sources télévisuelles. Déjà en 2015, [171] les CNN dépassaient 95% de justesse sur cette base de données, et les récentes avancées présentent des performances supérieures à 99% de justesse.

#### 3.5.2.2/ Mesures relatives au doigt

**Empreintes Digitales**

La vérification des empreintes digitales est surtout basée sur une comparaison directe entre deux empreintes et non sur une classification. Elle suit tout de même la logique de la chaîne habituelle de prétraitements mais une attention supplémentaire y est portée. En effet, selon le système d'acquisition utilisé (caméra, encre, capteur thermique, etc) ainsi que l'état de la peau du sujet (cicatrice, température extérieure, saleté, etc), les empreintes peuvent être partielles ou dégradées, et l'image peut contenir d'autres informations indésirables. Ainsi, comme illustré par la figure 3.17, ces prétraitements [172] consistent en une segmentation de la zone d'intérêt, une amélioration d'image (normalisation, filtrage, etc.), une binarisation (valeur de pixels : 0 ou 255) et une squelettisation (amincissement à 1 pixel de large). Certains travaux de la littérature [173] se penchent exclusivement sur l'amélioration d'image afin de reconstruire les parties manquantes ou dégradées.

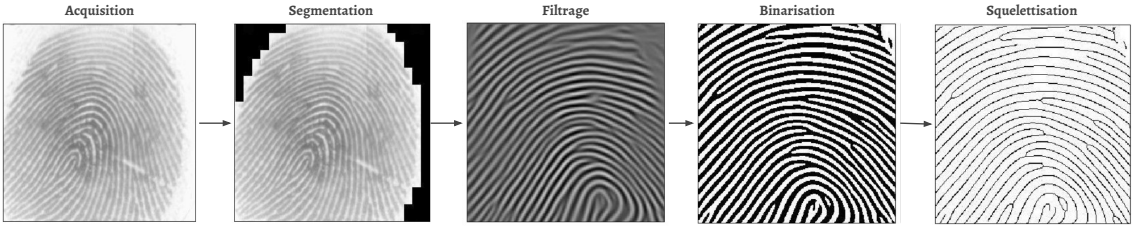



Figure 3.17 – Illustration de chaîne de prétraitement des empreintes digitales

Le processus d'extraction de caractéristiques des empreintes digitales permet d'obtenir des caractéristiques jusqu'à trois niveaux de précision [174] : **le premier niveau** fournit des caractéristiques globales et correspond au motif formé par l'orientation des lignes au centre de l'empreinte (ces motifs, classés selon diverses catégories [175], seront exposées dans la section 5.1.1), **le deuxième**

**niveau** fournit des informations locales correspondant à un ensemble de bifurcations et de terminaisons des lignes formant des motifs appelés "minuties" (également exposées dans la section 5.1.1), et **le troisième niveau** est le niveau le plus fin (disponible seulement sur des images de haute qualité), comprenant des informations sur la fréquence des lignes (largeur et espace entre les lignes) ou encore les pores de la peau. Il existe un grand nombre d'approches de correspondances d'empreintes digitales qui peuvent être classées selon trois catégories [176] : correspondance basée sur les minuties, basée sur les caractéristiques des crêtes papillaires, et basée sur un calcul de corrélation. Selon leur popularité dans la littérature, ces catégories peuvent également être réunies en deux grandes familles : les approches basées sur les minuties, et celles non basées sur les minuties.

**Approches basées sur les minuties [177] :**
L'approche basée sur les minuties est la plus populaire dans la littérature et comprend donc le plus de contributions. Elle consiste à trouver un alignement optimal entre deux modèles, construits à partir des minuties de deux images, duquel résulte un nombre maximal de correspondances entre les minuties. Parmi ces méthodes, on peut citer celles basées sur la transformée de Hough [178], les méthodes basées sur un regroupement de minuties [179] ou encore la correspondance hiérarchique de minuties [180]. Ces méthodes sont classées comme globales ou locales, puis alignées ou non.

- **Correspondance globale des minuties :** Les algorithmes de cette catégorie abordent le processus d'alignement entre les deux modèles en prenant en compte l'ensemble des minuties. Ils peuvent parfois avoir recours à une étape de pré-alignement basée sur d'autres caractéristiques telles qu'une carte d'orientations des crêtes, ou leurs fréquences.
- **Correspondance locale des minuties :** Elle porte sur les différentes structures locales des relations entre les minuties (graphe), telles que la distance entre des minuties au voisinage d'une autre. Ces méthodes sont invariantes aux translations et aux rotations, et permettent de trouver une correspondance même sur une image partielle. Ces méthodes peuvent être complétées en utilisant des caractéristiques locales telles que la texture.

**Approches non basées sur les minuties [181, 182] :**
Ces approches mesurent le degré de similarité via les intensités des pixels. Elles appliquent diverses transformations aux images et calculent un score de correspondance. Cette opération est sujette à des faiblesses liées aux variations de contraste, de lumière et aux déformations dans l'image. Dans la littérature, il existe des alternatives pour pallier à certains de ces problèmes [177]. L'utilisation de fenêtres locales autour des minuties [183] ou l'alignement de singularités avant l'opération de corrélation [184] permettent d'atténuer les problèmes de distorsion. Le calcul de la corrélation par régions locales dans le domaine de Fourier [185] permet de réduire la complexité des calculs, et l'invariance en translation ainsi qu'en rotation est obtenue en utilisant la transformée de Fourier-Mellin [186]. D'autres approches n'utilisent pas directement les minuties [187], comme par exemple les *FingerCode* [188] utilisant l'information de texture par zones.

Les CNN ne sont pas en reste, et sont beaucoup utilisés afin d'effectuer une segmentation précise dans le but d'extraire des empreintes, parfois partielles, sur des images difficiles où les empreintes sont mélangées à d'autres informations (empreinte sur un magazine). Afin de segmenter une image d'empreintes, Dai et al. [189] entraînent un CNN à partir de patchs de textures d'empreintes digitales et d'autres de fonds quelconques. Tang et al. proposent un CNN à l'architecture spécifique [190], capable d'isoler une empreinte digitale d'une image très perturbée et d'en extraire les minuties. Afin d'améliorer les performances de correspondance dans le cas d'une identification de personne, Peralta et al. proposent un CNN [191] effectuant une première classification d'empreintes digitales à partir de caractéristiques de niveau 1. En comparaison, Pandya et al. [192] proposent d'effectuer une classification d'empreintes digitales parmi 56 personnes, basée sur les travaux de Bharadi et al. [193] et utilisant des images filtrées et squelettisées pour l'apprentissage d'un CNN au lieu d'un classique K-NN, augmentant alors la justesse de 77% jusqu'à 98,2%.

### Réseau vasculaire

Le réseau vasculaire présente la particularité d'être une modalité sous cutanée. Son acquisition s'effectue dans le domaine de l'infrarouge. En effet (voir §5.1), l'hémoglobine et l'oxyhémoglobine (au même titre que l'eau) présents dans le sang absorbent les longueurs d'ondes de l'infrarouge qui se trouvent disposer de la capacité à pénétrer la peau sur une certaine profondeur. Ainsi, la projection d'un éclairage infrarouge permet l'observation directe des veines. Celles-ci apparaissent sombres à l'image, qu'elles soient acquises par un capteur CMOS ou un capteur infrarouge (InGaAs, etc). Cette procédure particulière entraine cependant des difficultés de mesures et d'exploitations de l'image qui est souvent bruitée avec un très faible contraste. Ces difficultés sont dues aux nombreuses interactions avec la lumière telles que la diffusion interne [194], l'illumination inégale [195], ou encore à l'épaisseur des tissus relative au type de peau (doigt, dos de la main, avant-bras). De ce fait, comme pour les empreintes digitales, une grande proportion des études de la littérature concerne les méthodes permettant d'améliorer l'image en vue d'une segmentation optimale des veines.

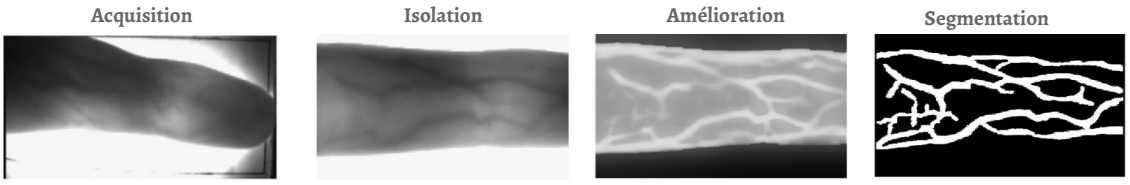



FIGURE 3.18 – Illustration de chaîne de prétraitement du réseau vasculaire du doigt [196]

Dans le but d'atténuer les effets d'un éclairage inégal, Kim et al. [195] soustraient à l'image sa composante d'éclairage, obtenue par la convolution locale d'un filtre moyenneur. L'histogramme de l'image normalisée résultante est égalisé afin d'améliorer la visibilité des veines. La correspondance entre deux images normalisées est alors effectuée à l'aide des descripteurs SIFT. En considérant la "propagation" des veines dans différentes directions, Yang et al. [197, 198] proposent l'utilisation de différents filtres orientés (banque de filtres de Gabor) afin d'augmenter la visibilité du réseau vasculaire, suivi d'une segmentation des veines [199] basée sur les K-moyennes et le concept du "*Image Matting" [200].*

Afin de réduire dans l'image l'influence des dispersions lumineuses dues à la peau, Lee et al. introduisent une méthode de restauration d'image de veines [201] et Yang et al. proposent un modèle optique biologique estimant ces dispersions [194]. Cependant les différences de structures de la peau, du muscle ou encore de l'os entre différentes zones du doigt peuvent mener à des images de mauvaise qualité [202]. À partir de ce constat, Yang et al. [203] proposent une méthode de restauration d'image considérant la structure de la peau et mettant en œuvre une PSF gaussienne et deux PSF de profondeur (réponse impulsionnelle spatiale ou *point spread function - PSF*). Dans le but d'éliminer ces variations de lumière dues à ces différences de structures, Xie et al. [204] proposent comme méthode de normalisation un rétinex à échelle unique basée sur un filtre guidé (*Guided Filter Based Single Scale Retinex - GFSSR*). Dans cette méthode, le rétinex simple échelle (*Single Scale Retinex - SSR*) [205] est complété par un filtre "guidé" [206] permettant de lisser l'image grâce à ses capacités ajustables de lissage tout en préservant les contours. Appliquée au réseau vasculaire de la rétine mais utilisable dans d'autres application, Fu et al. [207] proposent une segmentation du réseau vasculaire en utilisant une structure profonde nommée Deep Vessel. Cette structure met en œuvre un CNN à plusieurs échelles afin d'apprendre une représentation hiérarchique riche, et un étage de champs aléatoires conditionnels (*Conditional Random Fields - CRF)* [208] afin de modéliser les interactions entre les pixels à longue portée.

Il existe de nombreuses manières d'extraire les caractéristiques du réseau vasculaire et d'effectuer une correspondance entre deux images de veines. La présentation et la comparaison d'un très grand nombre de méthodes de chacune des étapes de la reconnaissance de veines sont fournies par Shaheed et al. dans leur étude complète de l'état de l'art [209]. D'après cette étude, les techniques d'extractions de caractéristiques du réseau vasculaire peuvent être classées selon quatre groupes : les méthodes basées sur les veines, basées sur des minuties, basées sur un modèle binaire local et basées sur la réduction de dimensions.

— **Méthodes basées sur les veines :** Parmi ces méthodes on retrouve des techniques basées sur le filtre de Gabor tel que Yang et al. [198], page précédente, avec une banque de filtres selon huit orientations. On retrouve aussi des techniques de suivi de lignes, comme proposé par Lemaitre et al. [210] avec leur détecteur de structures curvilignes ou encore Bekkers et. al. [211] effectuant un suivi complet de la structure du réseau vasculaire rétinien.
— **Méthodes basées sur des minuties :** D'une manière similaire aux empreintes digitales, les minuties ici constituent les points aux embranchements (ou bifurcations) et les jonctions des vaisseaux sanguins ainsi que leurs terminaisons. Leur détection avec une distinction entre ces différents types de "croisements" de lignes permet d'obtenir une information plus riche. De nombreux travaux portent sur la détection de ces points pour des images rétiniennes tel que [23, 212].
— **Méthodes basées sur le motif binaire local :** L'algorithme LBP est utilisé ici en tant que descripteur permettant de représenter les caractéristiques locales des veines. Rosdi et al. [213] introduisent l'approche "LLBP" (*Local Line Binary Pattern*), basée sur le LBP, extrayant l'information dans les directions horizontale et verticale. Lu et al. proposent une version particulière du "LLBP", le "GLLBP" (*Generalized Local Line Pattern*) [214], où l'orientation du modèle de ligne est sélectionnable dans n'importe quelle direction. Liu et al. [215] introduisent une nouvelle version nommée "CLLBP" (*Customized Local Line Binary Pattern*), permettant d'augmenter le pouvoir discriminant de l'information des caractéristiques locales et de diminuer le temps de calcul de la correspondance.
— **Méthodes basées sur une réduction de dimension :** Avec ces méthodes, basées sur l'algorithme ACP, les images de réseaux vasculaires sont projetées dans un sous espace de dimensions réduites. La plupart de ces techniques nécessitent une phase d'apprentissage afin de déterminer le sous espace optimal. Haijian et al. [88] utilisent la technique de réduction ACP munie d'un noyau, "KPCA", afin d'obtenir un compromis entre pertinence des caractéristiques (ainsi que leur nombre) et vitesse d'extraction de ces caractéristiques.

Selon la récente étude de Shaheed et al. [209], les méthodes de correspondances d'images du réseau vasculaire peuvent être réunies sous trois catégories. D'une part les méthodes conventionnelles, basées sur des mesures de distances, atteignant jusqu'à 97,8% de justesse sur un problème d'identification, mais étant relativement sensibles au bruit. D'autre part les méthodes du Machine Learning traditionnel, utilisant des classifieurs tels que les SVM ou les K-NN et approchant les 100% de justesse pour les meilleures. Et pour finir, les méthodes utilisant le Deep Learning et principalement des CNN, sont plus stables et relativement proches d'une justesse maximale.

#### 3.5.2.3/ Méthodes sélectionnées

L'objectif du projet dans lequel s'inscrit cette thèse consiste à sécuriser l'accès à une zone restreinte. Nous avons choisi de nous orienter sur un système de sécurité biométrique basé sur l'image, répondant aux diverses normes, contraintes et recommandations des autorités chargées d'encadrer l'utilisation de tels dispositifs. Nous proposons de traiter un système multi-biométrique, utilisant d'une part le visage et d'autre part les empreintes digitales ainsi que le réseau vasculaire du doigt. L'étude présentée dans ce chapitre traitant de la littérature relative à ces modalités nous permet de restreindre notre étude à certaines méthodes.

Ainsi, pour le visage, nous utilisons l'algorithme de Viola-Jones afin de détecter et d'extraire la région d'intérêt du visage, du fait de sa rapidité de mise en place et de traitement. Le système étant fixe, la faiblesse de cet algorithme quant à l'orientation du visage n'est pas particulièrement contraignante. Dans le but de réduire les perturbations extérieures et d'acquisitions, nous proposons d'observer l'influence de deux types de filtres que nous expliciterons dans le chapitre suivant : un filtre passe haut (filtre de Sobel) ainsi qu'une normalisation d'éclairage (filtre de TanTriggs). L'extraction de caractéristiques est effectuée par l'ACP et, dans le but de réduire la quantité d'informations biométriques, nous observerons plusieurs seuils du nombre de valeurs propres conservées. Enfin, pour effectuer la classification, nous proposons de comparer deux classifieurs du Machine Learning traditionnel : les SVM avec les RF. Dans le but faire correspondre cette étude avec l'état de l'art, des CNN légers seront utilisés, par le biais du Transfer Learning, en tant qu'extracteur de caractéristiques et leur couche Softmax sera ré-entraînée comme classifieur.
Motivés par le besoin d'accroître la sécurité en comblant les lacunes de l'authentification 2D du visage, l'objectif de notre seconde étude biométrique consiste à traiter de deux modalités du doigt, à savoir les empreintes digitales et le réseau vasculaire. Ces deux modalités seront exploitées avec une chaîne de traitements similaire. Celle-ci comporte une partie d'amélioration d'image, une partie segmentation/binarisation, une partie extraction de caractéristiques et une partie correspondances. De la même manière que pour les visages, l'amélioration d'image consiste à réduire l'influence des perturbations lumineuses et du bruit dans l'image. Une banque de filtres de Gabor est ainsi utilisée. Pour les empreintes digitales, les paramètres du filtre sont calculés à partir de l'orientation et de la fréquence des lignes par sous fenêtrage dans l'image. Pour les veines, plusieurs filtres de différentes orientations sont appliqués sur toute l'image et chaque pixel est sélectionné parmi les résultat des filtres selon une règle minimale. S'en suit une segmentation adaptative locale et une squelettisation. Les caractéristiques extraites sont formées des points formant les fins de lignes, les bifurcations/croisements et les lignes elles mêmes. Des descripteurs en ces points seront calculés avec l'algorithme ORB dérivant des SIFT et l'authentification est effectuée par un classifieur à partir de la correspondance brute (*"Brute Force Matching"*) entre deux images.

Dans une démarche fortement liée au profil industriel du projet de cette thèse, les algorithmes de ces différentes études de la modalité du visage seront implémentés dans une caméra intelligente intégrant tous les traitements. Les temps de calculs de chaque phase seront alors mesurés afin de pouvoir présenter une comparaison complète avec le Deep Learning (temps de calculs, performances brutes, taille de stockage des modèles), ces comparaisons étant jusqu'à maintenant [216] peu nombreuses dans la littérature. Un prototype original d'acquisition des empreintes digitales et du réseau vasculaire sera également présenté. A partir des résultats des études algorithmiques, les meilleures configurations seront retenues. Les performances du système complet seront alors mesurées avec la mise en place d'une fusion au niveau décision entre les prédictions du visage et soit, les prédictions des empreintes digitales et du réseau vasculaire séparément, soit les prédictions de leur fusion, soit les prédictions de leur fusion et des deux modalités indépendantes.

# 4

# Authentification de visages : comparaison de méthodes

Notre étude algorithmique est composée de deux chapitres, et présente les protocoles de développement ainsi que les performances d'authentification des modalités biométriques considérées. Ce chapitre traite de la phase d'authentification des personnes par leur visage en deux dimensions. Basée sur l'état de l'art précédent, cette étude fait appel d'une part au Machine Learning classique et d'autre part au Deep Learning, plus précisément aux réseaux de convolutions (CNN) via le "Transfer Learning".

Nous proposons ainsi une comparaison entre différentes méthodes sur une problématique identique. Bien que peu considérés dans la littérature mais important au regard de notre contexte applicatif, les critères traités par cette étude sont étroitement liés à la réduction de la taille de stockage du modèle biométrique mais aussi au respect d'une réponse en temps réel.

Dans une démarche de réduction de la quantité d'informations biométriques, nous comparons ces méthodes afin de mettre en lumière un compromis entre le nombre d'images d'apprentissage, les performances d'authentification et les tailles des modèles résultants. Afin d'aider à l'interprétation des résultats, les bases de données utilisées sont exposées ainsi que les différents prétraitements permettant de réduire l'impact de certaines perturbations liées aux protocoles de l'étude.

## Sommaire

## 4.1/ Protocoles de l'étude

### 4.1.1/ Données d'entrée et variabilités environnementales

#### 4.1.1.1/ Bases de données

L'authentification de visages relève d'une problématique d'apprentissage à deux classes, traduisant la question : "L'échantillon mesuré correspond-il au modèle stocké sur la carte sans contact ?". Ces classes distinguent d'un côté la personne disposant d'une autorisation d'accès et de l'autre des personnes non autorisées. Dans cette étude, elles sont composées d'images de visages provenant de diverses bases de données dont des échantillons sont présentés figure 4.1.

L'objectif est de conserver une certaine maîtrise des différentes variations du sujet ou de l'environnement, pouvant perturber l'acquisition, et de réaliser un démonstrateur temps réel. L'utilisation de la base de donnée "Le2i", construite au sein de notre laboratoire et présentée figure 4.1d, nous permet d'inclure des variations spécifiques dans l'image (contrôle des conditions d'acquisition), et de vérifier la condition du fonctionnement temps réel directement avec les sujets. Ainsi, la première classe généralisant les visages de la personne habilitée comprend des images issues de cette base. La seconde classe est composée de visages issus de différentes bases de données de la littérature ("Labeled Faces in the Wild" ou "LFW" figure 4.1a ; "Essex" ou "Face94" figure 4.1c), permettant de considérer dans les données une variabilité quant aux capteurs d'acquisition.
Les modèles résultant de la classification sont ensuite testés via une prédiction de classe sur de nouveaux échantillons inconnus. Dans la suite de ce chapitre, cette évaluation permet de caractériser les performances des modèles en effectuant diverses mesures (présentées dans le chapitre précédent) à partir des prédictions effectuées sur 1000 images indépendantes pour chacune des deux classes. Le taux de faux positifs (ou intrusions) est calculé sur les prédictions de la classe des personnes non autorisées, testée à partir d'échantillons provenant d'une combinaison des bases LFW, ESSEX et d'images complémentaires de la base Le2i. Le taux de faux négatif est bien sûr évalué à partir de nouveaux échantillons du sujet pour lequel le classifieur a été entrainé. La base

(a) Base de visages "LFW"

(b) Base de visages "AT&T-ORL"

(c) Base de visages "ESSEX-Face94"

(d) Base de visages "Le2i"

Figure 4.1 – Échantillons en niveaux de gris des différentes bases de données utilisées

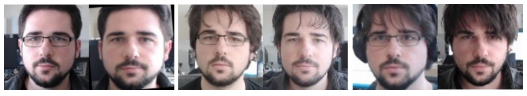
LFW [169] est composée de 5749 personnalités connues, réparties au travers de 12233 images acquises via un détecteur de visages sur des images télévisuelles, et donc en environnement non contraint. La base ESSEX [217] est formée de 3060 images, provenant de 153 sujets différents auxquels il a été demandé de parler lors l'acquisition en laboratoire sur un fond de couleur verte.

La base ORL [164] contient 400 images de 40 sujets distincts acquises en laboratoire. Notre base de donnée Le2i, construite sur plusieurs jours sous diverses contraintes via une caméra générique (webcam du commerce), se compose de 11 sujets distincts. Afin d'augmenter la quantité d'échantillons, les images ont été soumises à différentes variations de gamma, de contraste, ainsi qu'à des effets miroirs verticaux, permettant alors d'obtenir 33000 images au total. Ces bases de données présentent l'avantage de prendre en compte un grand nombre de variations que l'on peut trouver en conditions réelles et permettent, via l'utilisation de la base de données Le2i, de reproduire les expérimentations via un démonstrateur temps réel.

#### 4.1.1.2/ Variations environnementales et d'acquisition

Afin de garantir des performances d'authentification optimales, il est nécessaire de conserver une forte stabilité entre les différentes acquisitions du visage. Sans contraintes lors de la mesure (distance à la caméra, orientation du visage via un "patron" sur une interface utilisateur, etc), il est fort probable de trouver des différences, parfois majeures, entre deux acquisitions. La phase d'enrôlement (ou apprentissage) pouvant être effectuée dans un lieu particulier différent du lieu d'implantation (c.à.d. un lieu dédié à cette tâche sous le contrôle d'un administrateur), il est nécessaire de porter une attention particulière aux contraintes imposées à l'utilisateur. En effet, diverses variations, à la fois environnementales et intrinsèques au sujet, viennent troubler cette stabilité.

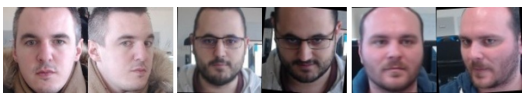
Outre celles nécessitant de plus amples contraintes d'acquisitions afin d'être maîtrisées, certaines peuvent être corrigées simplement et d'autres en partie atténuées. Plusieurs de ces variations, illustrées par des exemples figure 4.2, sont présentes dans la base de visages Le2i. On retrouve d'abord des variations temporelles figure 4.2a, induisant des différences propres au sujet et pouvant perturber la mesure si le modèle biométrique n'est pas régulièrement mis à jour. Les variations d'éclairage, figure 4.2b, dues à l'horaire ou à un dispositif d'éclairage non maîtrisé, introduisent de fortes variations en marquant les zones d'ombres ou en surexposant une zone du visage. Les deux derniers exemples, figures 4.2c et 4.2d, sont directement liés au comportement du sujet et peuvent être corrigés en lui imposant une contrainte lors de l'acquisition (e.g principe des cabines photographique). Bien que non présentes dans les bases de données utilisées, il est important de noter qu'il existe également des variations d'échelle liées à la distance entre la caméra et le sujet, pouvant être corrigées de la même manière que les variations comportementales. En effet, bien que le visage détecté soit rogné puis redimensionné à une taille fixe de $\omega \times \omega$ pixels, si le visage du sujet sur l'image représente $\Omega \times \Omega$ pixels et que $\Omega < \omega$, alors la qualité de l'image de la zone d'intérêt ainsi rognée se verra grandement réduite.



(a) Variations temporelles

(b) Variations d'éclairage

(c) Variations d'expression



(d) Variations d'orientation hors plan de l'image

Figure 4.2 – Exemples de variations perturbant la stabilité des mesures

### 4.1.2/ PROTOCOLES EXPÉRIMENTAUX

L'étude menée et présentée dans ce chapitre porte sur l'authentification de visages. D'une part, deux classifieurs propres au *Machine Learning* "classiques" sont comparés sous diverses configurations. D'autre part, afin de confronter ces méthodes à l'état de l'art, deux réseaux de convolutions sont entraînés sur nos images par transfert de connaissance (*Transfer Learning*). Comme illustré par la figure 4.3 détaillant les protocoles d'apprentissage et de test, ces méthodes sont comparées, indépendamment de la cible d'implantation, en termes de performances d'authentification et de taille de stockage des modèles biométriques. En effet, pour pouvoir être conservés sur un support individuel sans contact de type carte RFID, ces derniers doivent respecter un critère de taille de stockage précis. Afin d'aider à une sélection quant à la méthode la plus pertinente suivant la cible d'implantation, une étude complémentaire traitant des temps de calculs sur une cible embarquée est présentée en section 6.2.2 du manuscrit. Cette sélection est ensuite effectuée dans la section 6.1.2 du manuscrit, qui traite des performances du système global.

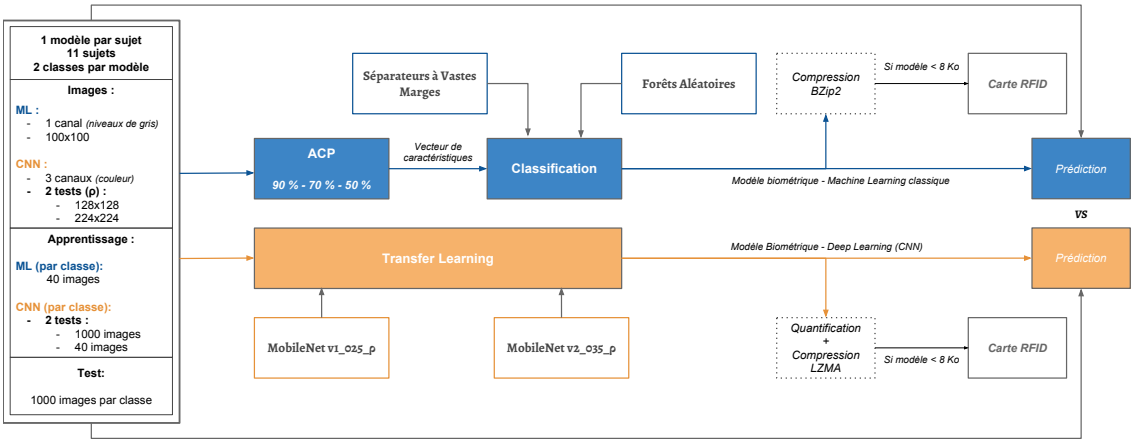
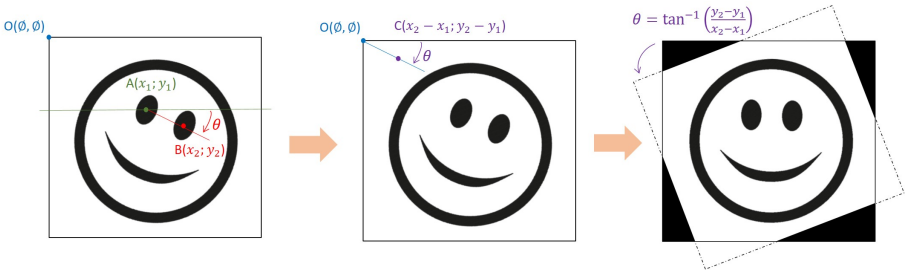
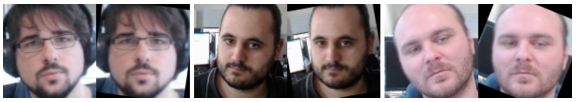



FIGURE 4.3 – Résumé de la comparaison Machine Learning Classique et Deep Learning

La première étape, commune aux méthodes comparées dans cette étude et nécessaire pour conserver un maximum de stabilité entre les mesures (répétabilité), consiste à réduire les variations entre les bases de données en normalisant l'orientation (ou inclinaison) horizontale des visages par un "alignement". Il est en effet possible, en utilisant les coordonnées des yeux dans le plan de l'image, de corriger la position du visage via une rotation en deux dimensions (rotation suivant de l'axe orthogonal à l'image) afin d'aligner les yeux horizontalement. Le principe de cet alignement ainsi que des exemples de résultats sont exposés figure 4.4. Considérant le point $A(x_1, y_1)$ comme étant l'œil gauche et le point $B(x_2, y_2)$ comme l'œil droit, le point $C(x_2 - x_1, y_2 - y_1)$ correspond à la position du point B par rapport au point A ramené à l'origine. L'angle $\theta$ entre les deux yeux peut alors être calculé via la formule $\theta = arctan(\frac{y_2-y_1}{x_2-x_1})$. Il suffit alors d'effectuer une rotation de cet angle $\theta$, ajoutant également dans les coins de l'image des zones noires du fait de l'inexistence de pixels à cet endroit.
Les différences d'orientations verticales du visage (en dehors du plan de l'image, donc en 3 dimensions) ne peuvent pas être corrigées. Certaines opérations telles que "l'homographie" peuvent en atténuer les effets via une modification de la perspective, mais ce type d'opération nécessite au préalable une calibration de la caméra. De plus, la partie du visage non visible à l'image ne pourra pas être retrouvée. Ces variations étant dépendantes du comportement du sujet, de la même manière que les variations d'expressions, elles ne peuvent être corrigées efficacement que part une contrainte lors de l'acquisition.




(a) Principe de redressement du visage



(b) Correction d'alignement des visages

FIGURE 4.4 – Correction d'alignement des visages

Quant aux variations d'éclairage, dans la mesure où elles ne sont pas trop importantes (i.e ne suppriment pas d'informations par des zones saturées), il est possible de les atténuer via une étape de filtrage. Divers filtres, présentés en section 4.2.1 et appliqués aux exemples d'images perturbées de la figure 4.2, seront alors expérimentés afin de réduire l'influence de ces variations et d'améliorer les performances des méthodes classiques de Machine Learning, lesquelles font usage de l'analyse en composantes principales qui est sensible à ce type de variations. Les CNN sont eux même composés de nombreux filtres dont les paramètres sont optimisés lors de l'apprentissage. Ils seront ainsi seulement soumis aux images d'origine seulement.

## 4.2/ MACHINE LEARNING CLASSIQUE : MÉTHODES ÉPROUVÉES

### 4.2.1/ PROTOCOLES ET MISE EN PLACE

Les premières méthodes exploitées dans ce chapitre appartiennent à la famille du Machine Learning classique. Afin de compenser certaines des variations exposées précédemment, elles mettent en œuvre des prétraitements, une étape d'extraction et de réduction de caractéristiques suivie d'un apprentissage par un classifieur. Différentes configurations de ces étapes sont évaluées dans le but de trouver un compromis entre la rapidité des traitements, la quantité d'informations à conserver (propres à la classification) et les performances d'authentification. De nombreux algorithmes peuvent assurer ces fonctions et, dans une démarche de recherche et d'évaluation de méthodes, deux prétraitements ainsi que deux classifieurs seront étudiés et comparés. Les résultats de cette étude sont présentés dans la section 4.2.2.

#### 4.2.1.1/ RÉPARTITION ET TRAITEMENT DES DONNÉES D'ENTRÉES

Les méthodes employées utilisent généralement en entrée des images de taille fixe ne disposant que d'un canal (niveau de gris). Des images de dimension $100 \times 100$ pixels nous paraissent conserver suffisamment d'informations relatives au visage pour être exploitables et représentent un point de départ intéressant dans la limitation de la complexité des données. La phase d'apprentissage utilise les caractéristiques de visages de quarante images par classe et, comme présenté précédemment, 1000 images par classe sont utilisées pour la phase d'évaluation. La quantité d'informations représentée par les images de visages est trop importante pour être directement utilisée par les classifieurs ($100 \times 100 = 10000$ valeurs par image). De plus, tous ces pixels ne représentent pas

forcément une information pertinente, il est donc nécessaire de passer par une étape d'extraction de caractéristiques propres aux visages. Dans cette étude, l'Analyse en Composante Principale (ACP) a été choisie pour remplir cette fonction. De nature sensible aux variations exposées précédemment, une phase de prétraitement s'impose. Deux filtrages proposés dans la littérature et souvent utilisées pour la reconnaissance de visages sont alors comparées : Sobel et TanTriggs.

**Filtre de Sobel :**

Le filtre de Sobel détecte les contours dans l'image (hautes fréquences) en localisant les variations maximales du gradient entre l'intensité d'un point et ses voisins. Afin d'éviter de détecter des artefacts non pertinents, un flou gaussien de noyau $3 \times 3$ est appliqué à l'image en niveau de gris. Puis, suivant l'équation 4.1, deux masques de taille $3 \times 3$ sont convolués en tout point de l'image résultante ($I_{Gauss}$), déterminant les variations verticales et horizontales d'intensité. La moyenne des images résultantes permet d'obtenir les contours de l'image d'origine.

$$G_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \qquad G_x = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \qquad I_{Sobel} = \frac{|G_x * I_{Gauss}| + |G_y * I_{Gauss}|}{2} \tag{4.1}$$

**Filtre de TanTriggs :**

Tan et Triggs [162] proposent une méthode de normalisation d'éclairage permettant de réduire les différences de luminosités entre les images. Le filtre de TanTriggs distingue trois étapes à partir d'une image en niveaux de gris "$I_{gris}$", pour lesquelles les paramètres "$\gamma$", "$\alpha$", "$\tau$", "$\sigma_0$" et "$\sigma_1$" sont réglables afin d'ajuster le filtre. Initialisés à partir des suggestions des auteurs puis modifiés graduellement afin d'obtenir des résultats visuellement optimaux sur nos images (peu de bruit et normalisation homogène entre des échantillons de faibles et de fortes luminosités), ces paramètres ont été fixés à $\gamma = 0,8$, $\tau = 10$, $\alpha = 0.6$, $\sigma_0 = 0,5$ et $\sigma_1 = 2$. En général, afin de conserver ses propriétés, il est judicieux d'ajuster la taille du noyau du filtre gaussien en fonction de la valeur de $\sigma$. En conséquence, la taille du noyau est déterminée via $(2.\lceil 3.\sigma \rceil + 1)$, donnant un noyau de 5x5 pour $G_x$ ($\sigma_0 = 0,5$) ainsi qu'un noyau de 13x13 pour $G_y$ ($\sigma_1 = 2$). Les différentes phases de ce filtre sont les suivantes :

**1. La correction gamma :**

$$I_{gamma}(x, y) = I_{gris}(x, y)^{\gamma}, \quad \text{où} \quad \gamma \in [0, 1] \tag{4.2}$$

Cette correction permet d'augmenter la dynamique locale des zones sombres de l'image, tandis qu'elle compresse celle des zones éclairées.

**2. La différence de gaussiennes (DoG) :**

$$\left.\begin{aligned} G_x &= \frac{1}{\sqrt{2\pi\sigma_0^2}}.e^{-\frac{x^2}{2\sigma_0^2}} \\ G_y &= \frac{1}{\sqrt{2\pi\sigma_1^2}}.e^{-\frac{y^2}{2\sigma_1^2}} \end{aligned}\right\} \Rightarrow I_{DoG}(x, y) = I_{gamma}(x, y) * G_x - I_{gamma}(x, y) * G_y \tag{4.3}$$

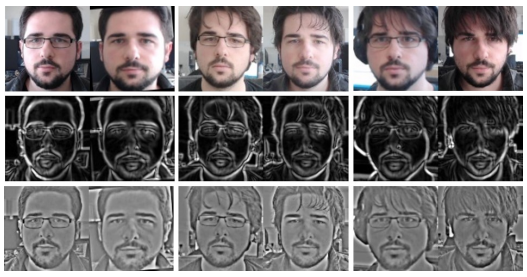
où $G_x$ et $G_y$ sont les distributions gaussiennes suivant respectivement l'axe $x$ et l'axe $y$, pour une taille de noyau donnée, et $\sigma_0/\sigma_1$ sont leur écarts type associés. L'image résultante, $I_{DoG}$, est la différence des convolutions de ces gaussiennes avec l'image corrigée "$I_{gamma}$".

**3. Égalisation de contraste :**

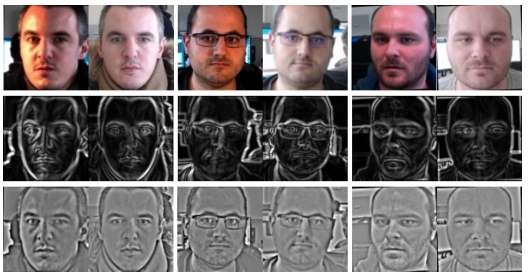
$$\left.\begin{aligned} I_1(x,y) &= \frac{I_{DoG}(x,y)}{(mean(|I_{DoG}(x,y)|^{\alpha}))^{\frac{1}{\alpha}}} \\ I_2(x,y) &= \frac{I_1(x,y)}{(mean(min(\tau,|I_1|)^{\alpha}))^{\frac{1}{\alpha}}} \end{aligned}\right\} \Rightarrow I_{TanTriggs}(x,y) = \tau.\tanh\left(\frac{I_2}{\tau}\right) \tag{4.4}$$

Cette phase rééchelonne les intensités lumineuses via deux premières étapes pour lesquelles une approximation de la variation de contraste est effectuée, où $\alpha$ est un paramètre réduisant significativement l'influence des hautes valeurs, $\tau$ est un paramètre permettant de seuiller par la suite ces grandes valeurs et la moyenne est calculée sur l'image entière. L'image $I_2$ résultante est bien seuillée mais présente toujours des valeurs extrêmes. La dernière phase consiste à réduire une dernière fois leur influence via une fonction non linéaire.

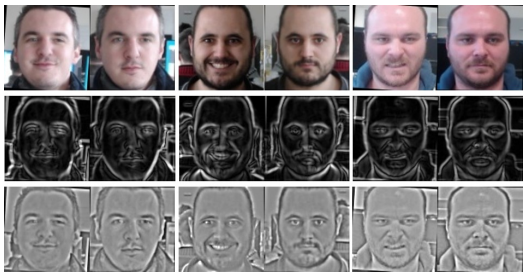
Afin d'en visualiser l'impact, la figure 4.5 illustre les images résultant de ces filtrages sur les exemples de variations précédemment présentées. La deuxième ligne de chaque mosaïque d'images correspond au filtre de Sobel et la troisième correspond au filtre de TanTriggs, avec les paramètres explicités dans la présentation de ces filtres. La figure 4.5b démontre l'efficacité de ces prétraitements quant à la correction des variations d'éclairage. En revanche, ils ne permettent pas de corriger les variations liées aux traits physiques des sujets. A la différence du filtre de Sobel qui n'extrait que les contours, le filtre de TanTriggs normalise l'intensité lumineuse entre les images et permet alors de conserver plus d'informations.



(a) Variations temporelles



(b) Variations d'éclairage



(c) Variations d'expressions

(d) Variations d'orientation

FIGURE 4.5 – Illustration des filtres appliqués aux diverses variations.
Par ligne : Images originales ; Filtre de Sobel ; Filtre de TanTriggs

Néanmoins, les images résultantes restent très différentes des originales, bien que les variations d'éclairage aient été atténuées. De ce fait, et pour mesurer précisément l'impact de ces filtres sur les performances d'authentification, cette étude se propose d'ajuster graduellement la contribution de ces prétraitements via une somme pondérée. En effet, suivant l'équation 4.5, l'image résultante, $I_{Fusion}$, est composée à $[\beta]$ % de l'image filtrée, $I_{Filtre}$, et à $[100 - \beta]$ % de l'image en niveaux de gris associée, $I_{Gris}$. L'impact de ces filtrages est alors observé pour différentes valeurs de $\beta$, fixées à : $\beta = \{0, 20, 40, 60, 80, 100\}$. La figure 4.6 illustre ces différentes images, et leur impact sur les performances d'authentification est présenté dans la section 4.2.2.

$$I_{Fusion} = \frac{(100 - \beta)}{100}.I_{Gris} + \frac{\beta}{100}.I_{Filtre}, \qquad \beta \in [0; 100] \tag{4.5}$$


(a) Échantillons $I_{Fusion}$ - Sobel (b) Échantillons $I_{Fusion}$ - Tan-Triggs

FIGURE 4.6 – Échantillons des filtres appliqués à chaque base suivant les valeurs de $\beta$

#### 4.2.1.2/ EXTRACTION ET RÉDUCTION DES CARACTÉRISTIQUES

L'extraction de caractéristiques est une étape cruciale. En effet, si les pré-filtrages présentés précédemment permettent d'améliorer les performances d'authentification en compensant les variations d'acquisitions, l'extraction de caractéristiques est indispensable pour la classification et permet de considérer uniquement les éléments pertinents en décrivant de manière optimale l'image via des caractéristiques propres aux visages. En plus d'isoler ces attributs, cette phase permet de réduire grandement la quantité d'informations à traiter. Les méthodes présentées ici utilisent l'Analyse en Composante Principale (ACP ; *PCA en anglais*) pour décrire ces éléments. Cette méthode, décrite chapitre 3, est la base de la méthode reconnue des "Eigenfaces" [117]. Elle utilise un certain nombre d'images dont les caractéristiques communes forment un espace propre de visages (matrice de vecteurs propres). Dans cette étude, l'ACP a été effectuée à partir de l'intégralité de la base de visages ORL, figure 4.1b, produisant alors un espace de visages à 400 dimensions ou caractéristiques (correspondant au nombre d'images dans la base de visages).

Que ce soit pour l'apprentissage ou la phase d'authentification, chaque échantillon (ou région d'intérêt rognée et redimensionnée à 100x100 pixels) y est ensuite projeté afin d'obtenir un vecteur composé des poids issus d'une combinaison linéaire entre ces caractéristiques. Afin de répondre à la problématique de réduction de la complexité des calculs, au même titre que la réduction des temps de calculs et de la quantité des données biométriques, il a été envisagé d'exploiter un seuillage du nombre de vecteurs propres issus de l'ACP. Dans la littérature, l'ACP utilise souvent un nombre restreint ($K$) de vecteurs propres via un seuillage ($\tau$) à 95% ou 90% du cumul des contributions (ou inertie) des valeurs propres associées ($\lambda$), respectant l'équation 4.6, où $N$ est le nombre total de valeurs propres. L'objectif étant d'étudier précisément l'influence de cette réduction, nous proposons d'appliquer différents seuils de plus en plus restrictif, à partir de ceux communément utilisés. Trois seuils donc ont été sélectionnés : 90%, 70% et 50% de l'inertie des valeurs propres. Le vecteur résultant de la projection d'une nouvelle image est donc composé d'autant d'éléments qu'il y a de valeurs propres conservées.

$$\sum_{i=1}^{K}(\lambda_i) \geqslant \tau . \sum_{i=1}^{N}(\lambda_i) \tag{4.6}$$

L'étape de filtrage modifie l'aspect de l'image, or l'extraction de caractéristiques en dépend. Pour exploiter ces images résultantes, il est alors nécessaire d'appliquer l'ACP sur les images issues du filtrage et donc créer un espace propre "filtré". Considérant alors trois seuils pour l'ACP, deux filtres ainsi que cinq valeurs différentes de $\beta$ en plus des images non filtrées, cette étude traite 33 configurations distinctes par sujet et par classifieur. Pour chacune de ces configurations d'ACP, l'espace propre associé étant le même pour chaque sujet et ne contenant pas de données à caractère privé, il présente l'avantage majeur de pouvoir être conservé localement dans la mémoire du système, et cela en conformité avec les exigences de la CNIL.

#### 4.2.1.3/ Classifieurs employés

Après l'extraction de caractéristiques des images prétraitées, les vecteurs résultants sont utilisés par un classifieur pour effectuer soit un apprentissage, soit une prédiction de classe (authentification). Lors de la phase d'entraînement, les classifieurs utilisent une matrice contenant, pour chaque classe, les vecteurs de caractéristiques des visages dédiés à l'apprentissage, auxquels est associé un vecteur contenant les "labels" des classes pour chacun d'eux. Ici, avec deux classes et 40 images par classe ($X$ = *nombre d'échantillons* = *2x40* = *80*), cette matrice d'apprentissage est de dimension 80x$\lambda$ ($X$x$\lambda$) et le vecteur des labels est composé de 80 éléments (un label par image).

Dans cette étude, deux classifieurs sont comparés : les Forêts Aléatoires (RF - *Random Forest*) et les Séparateurs à Vastes Marges (SVM - *Support Vector Machine*). Pour chaque sujet et à partir de chaque configuration d'ACP, ces classifieurs partagent la même matrice d'apprentissage ainsi que les mêmes échantillons de tests. Pour chacun de ces classifieurs, les paramètres d'apprentissage, détaillés dans les tableaux 4.1a et 4.1b, ont été utilisés pour l'intégralité de l'étude.
Les SVM, en apprentissage automatique, utilisent donc un noyau RBF, et les valeurs optimales des paramètres $C$ et $\gamma$ sont sélectionnées à chaque apprentissage parmi une grille de valeurs. La plage de valeurs correspondant à cette grille a été fixée au préalable après un certain nombre de tests sur nos données. L'apprentissage automatique est effectué via une validation croisée d'ordre 5 et le critère d'arrêt est fixé sur l'erreur $\epsilon$.

Pour les RF, seuls les critères d'arrêt de la croissance d'un arbre et de l'apprentissage ont été fixés (l'erreur OOB ainsi que les limites du nombre de branches et de leur profondeur). Le nombre de vecteurs de caractéristiques aléatoirement sélectionnés pour créer un arbre est définit comme étant la racine du nombre de vecteurs d'apprentissage.

| SVM | |
|---|---|
| C | Grille : $[2^{-1} ; 2^{8}]$ |
| $\gamma$ | Grille : $[2^{-30} ; 2^{-10}]$ |
| $\epsilon$ | 0,001 |
| Validation croisée | K-fold : 5 |

(a) Paramètres d'apprentissage des SVM

| RF | |
|---|---|
| Random vectors | $\sqrt{X}$ |
| Max trees | 150 |
| Max depth | 4 |
| Out Of Bag error | 0,001 |

(b) Paramètres d'apprentissage des RF

TABLE 4.1 – Paramètres d'apprentissage des classifieurs utilisés

#### 4.2.1.4/ RÉSUMÉ DES PROTOCOLES MIS EN ŒUVRE

Cette étude présente de nombreuses étapes que ce soit pour l'apprentissage ou la phase de test des classifieurs pour l'authentification de visages. Certaines de ces étapes (à savoir le filtrage des images de visages, la création de l'espace de visages (ACP), la réduction de la quantité d'information biométrique et l'apprentissage) sont étudiées sous différentes configuration. Afin de faciliter la compréhension de ces différents paramétrages, nous proposons de les résumer sous la forme de trois schémas : création de l'espace de visage (ACP), apprentissage des classifieurs et évaluation.

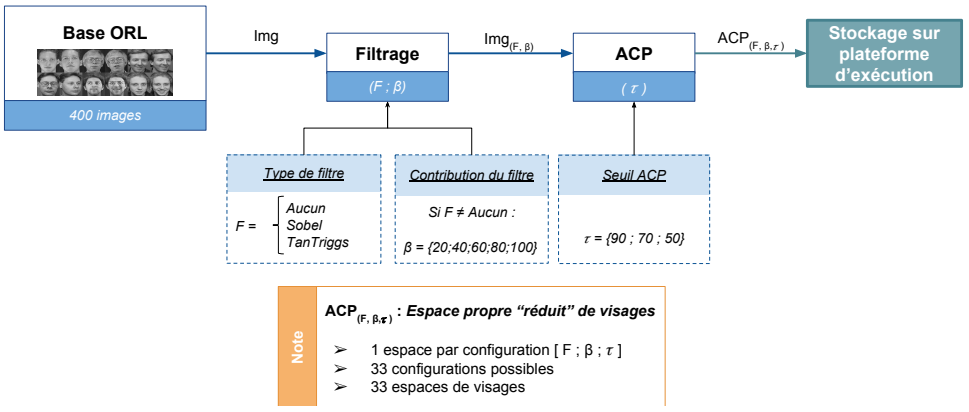



FIGURE 4.7 – Protocole de création de l'espace de visage par l'ACP

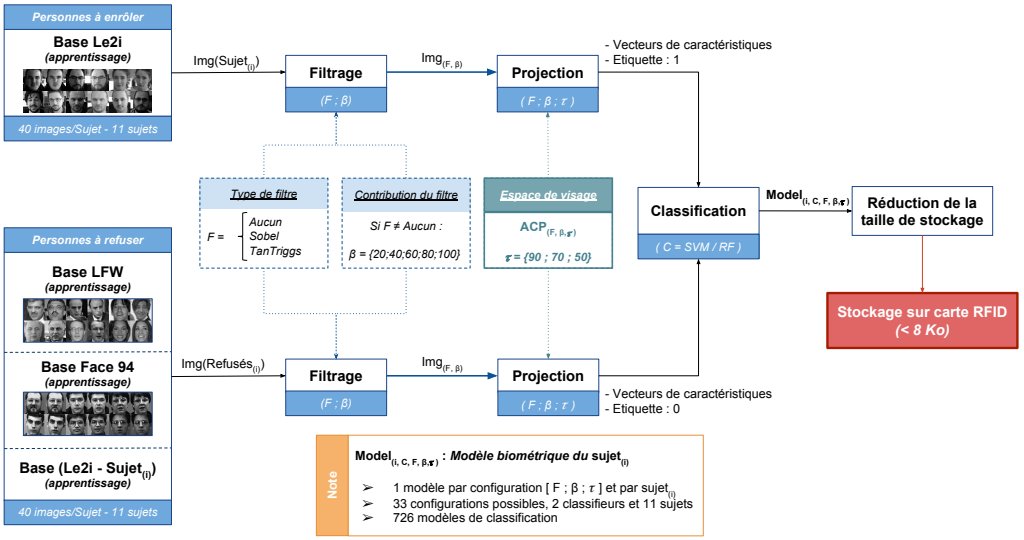



FIGURE 4.8 – Protocole d'apprentissage des classifieurs

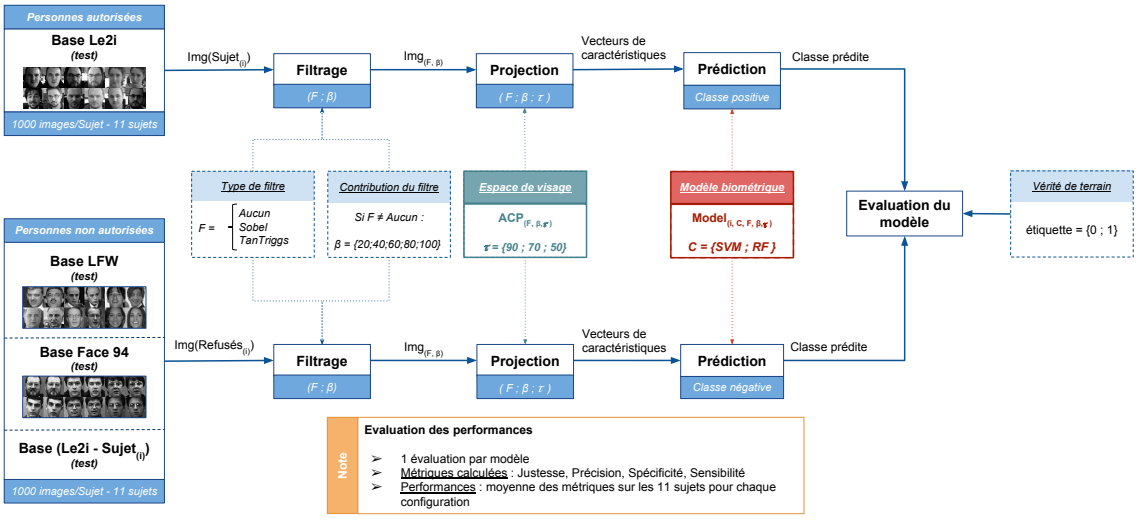



FIGURE 4.9 – Protocole de test et d'évaluation des classifieurs

### 4.2.2/ RÉSULTATS ET ANALYSE

Les classifieurs ont été entraînés suivant les protocoles exposés précédemment. Un espace de visage a été créé pour chaque configuration de filtre et de seuil de l'ACP. Afin de faciliter la compréhension des résultats, le tableau 4.2 présente le nombre de valeurs propres conservées ($\lambda$) pour chacune de ces 33 configurations, ainsi que le pourcentage que ce nombre représente par rapport au total de valeur propre ($N = 400$, correspond au nombre d'images utilisées pour créer l'espace propre). L'analyse du tableau 4.2 permet de remarquer que sans utilisation de filtre (ligne "pas de filtrage") le seuillage du nombre de valeurs propres est très efficace pour réduire les dimensions de l'espace propre. Afin d'expliquer l'influence du seuillage sur le nombre de valeurs propres de l'espace de visage, la figure 4.10 illustre par une courbe l'évolution du cumul de la contribution des valeurs propres de l'espace de visage créé avec des images non filtrées. Cette courbe met en évidence le nombre de valeurs propres nécessaires afin de représenter une certaine quantité (seuil de l'ACP) du cumul des contributions de toutes les valeurs propres.

| Filtre | ACP 90% | | ACP 70% | | ACP 50% | |
|---|---|---|---|---|---|---|
| | $\lambda$ | $\lambda/N$ [%] | $\lambda$ | $\lambda/N$ [%] | $\lambda$ | $\lambda/N$ [%] |
| Pas de filtrage | 112 | 28 | 22 | 5 | 7 | 2 |
| Sobel, $\beta$ = 20 | 128 | 32 | 25 | 6 | 7 | 2 |
| Sobel, $\beta$ = 40 | 174 | 43 | 45 | 11 | 11 | 3 |
| Sobel, $\beta$ = 60 | 221 | 55 | 85 | 21 | 29 | 7 |
| Sobel, $\beta$ = 80 | 240 | 60 | 108 | 27 | 45 | 11 |
| Sobel, $\beta$ = 100 | 240 | 60 | 109 | 27 | 46 | 11 |
| TanTriggs, $\beta$ = 20 | 168 | 42 | 40 | 10 | 10 | 2 |
| TanTriggs, $\beta$ = 40 | 224 | 56 | 80 | 20 | 22 | 5 |
| TanTriggs, $\beta$ = 60 | 266 | 66 | 130 | 32 | 55 | 14 |
| TanTriggs, $\beta$ = 80 | 290 | 72 | 165 | 40 | 87 | 22 |
| TanTriggs, $\beta$ = 100 | 299 | 75 | 178 | 44 | 99 | 25 |

TABLE 4.2 – Nombre de valeurs propres conservées ($\lambda$) après seuillage de l'ACP : L'ajout de filtres limitant l'influence de la variabilité des classes (luminance, etc) induit une augmentation du nombre de vecteurs propres nécessaires pour représenter une inerte donnée.

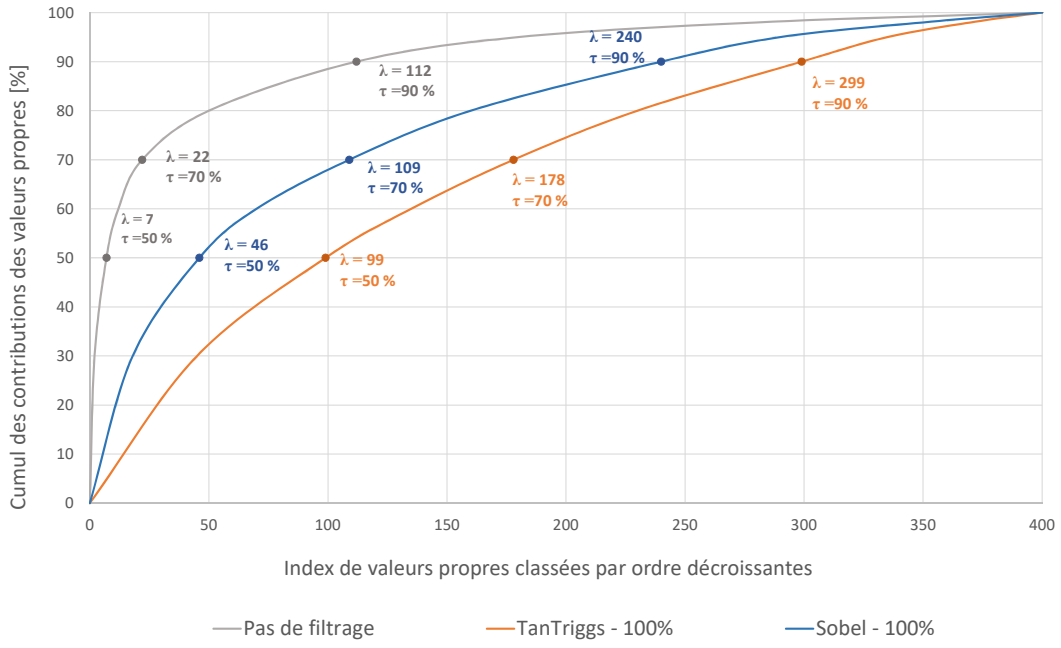



FIGURE 4.10 – Courbe du cumul des contributions des valeurs propres selon leur ordre décroissant, avec les trois seuils d'ACP étudiés, dans les cas où les images ne sont pas filtrées, filtrées totalement avec Sobel et filtrées totalement avec TanTriggs

Tronquer le nombre de valeurs propres à 90% du cumul de leur contribution totale réduit donc le nombre d'éléments de 400 ($N$) à seulement 112 (28% de $N$) dans le cas des images non filtrées. Le nombre de valeurs propres conservées chute ensuite très rapidement avec des seuils plus restreints : considérer 70% de l'information fournie par ces valeurs/vecteurs propres permet de s'affranchir de 90 éléments supplémentaires et on remarque que les 7 premières valeurs propres (2% de $N$) suffisent à représenter à elles seules 50% de l'information. Les vecteurs propres associés aux plus grandes valeurs propres représentent des caractéristiques générales des visages (basses fréquences), et ceux associés aux plus petites représentent des caractéristiques plus précises (hautes fréquences). Les filtres choisis réduisent les basses fréquences de l'image tout en augmentant la contribution des hautes fréquences. Ces nouvelles caractéristiques issues de l'ACP avec les images filtrées ne représentent donc pas l'information de la même manière que celles provenant des images non filtrées. Par conséquent, pour un seuil donné, l'utilisation d'un filtre requiert nécessairement plus de vecteurs propres pour représenter la quantité d'information descriptive correspondant à ce seuil.

Dans la suite de cette section, les résultats présentés considèrent directement la classe prédite par le classifieur : aucun seuil n'est appliqué sur les scores associés aux prédictions des classifieurs. Par conséquent, la représentation conventionnelle du couple sensibilité/spécificité par une courbe ROC n'est pas traitée ici. Cependant, une telle représentation de nos résultats est disponible dans la première formalisation de nos travaux [218]. Les valeurs de sensibilité, de spécificité et de précision de chaque classifieur pour chaque configuration, sont donc présentés dans les tableaux 4.3 et 4.4. Une série de graphiques, figure 4.11, représente ensuite la justesse d'authentification des modèles en correspondance avec les tailles de stockage mesurées d'une part directement après l'apprentissage, et d'autre part après leur réduction, comprenant une étape de réduction de précision des paramètres des classifieurs suivie d'une compression de ces derniers. Pour chaque configuration, ces résultats représentent une moyenne des 2000 prédictions (1000 par classe) sur chacun des 11 sujets testés (soit 22000 prédictions).

Afin d'observer rapidement les informations importantes des tables 4.3 et 4.4, certaines valeurs ont été mises en couleur. Le bleu représente les meilleures valeurs et le rouge les moins bonnes, parmi les deux types de filtres et les trois seuils d'ACP. Les valeurs en gras pour chaque couleur représentent les extrêmes pour chaque seuil d'ACP sans distinction du type de filtre.

On observe tout d'abord que l'utilisation du filtre de TanTriggs donne à la fois les meilleurs résultats ainsi que les moins bons, selon la valeur de $\beta$. Cependant, on peut noter que les résultats des deux filtres sont proches. Pour un seuil donné d'ACP, quel que soit le type de filtre, on remarque également que la différence entre les deux valeurs extrêmes d'une métrique d'évaluation est toujours inférieure à 11%. Les hautes valeurs de spécificité des RF (supérieures à 90%) nous indiquent que le système possède de bonnes capacités à "détecter" les personnes non autorisées (peu de faux positifs). Les valeurs de sensibilité des RF, plus faibles ici (inférieures à 90%), indiquent que le système aura plus de difficultés à reconnaitre la personne autorisé (quelques faux négatifs). Cependant, la sensibilité étant supérieure à 80%, nous pouvons établir que les capacités du système restent convenables. Pour finir, les valeurs de la précision des RF nous montrent que, parmi toutes les personnes acceptées par le classifieur, très peu font parties des faux positifs.
En comparaison, les valeurs de sensibilité des SVM sont plus proches des 90%, et les valeurs de sa spécificité ainsi que sa précision sont légèrement supérieures à celles des RF. Les SVM permettent donc d'obtenir un système avec moins de faux positifs et de faux négatifs. Dans les deux cas, le système peut donc être considéré comme étant restrictifs, en terme de sécurité, ce qui convient parfaitement au contexte de notre application.

L'observation des plus hautes valeurs du tableau nous donne une indication quant au comportement du système relativement au seuil d'ACP et à la contribution $\beta$ du filtre utilisé. En effet, diminuer le seuil d'ACP implique nécessairement d'augmenter la contribution $\beta$ du filtrage afin de conserver des résultats optimaux. Cette remarque est confirmée par l'étude de la justesse, présentée dans les graphes de la figure 4.11 suivant la présentation des résultats de réduction de la taille de stockage.

| **Filtre** | **ACP - 90%** | | | **ACP - 70%** | | | **ACP - 50%** | | |
|---|---|---|---|---|---|---|---|---|---|
| | **Se.** | **Sp.** | **Pr.** | **Se.** | **Sp.** | **Pr.** | **Se.** | **Sp.** | **Pr.** |
| Pas de filtrage | 86,9 | 95,5 | 95,1 | 83,9 | 94,1 | 92,4 | 79,9 | **84,9** | **84,1** |
| Sobel, $\beta$ = 20 | 87,3 | 96,1 | 95,7 | 83,9 | 93,1 | 92,4 | 77,3 | 89,6 | 88,2 |
| Sobel, $\beta$ = 40 | 86,1 | 95,0 | 94,5 | 86,0 | 94,5 | 94,0 | 75,6 | 92,0 | 90,4 |
| Sobel, $\beta$ = 60 | 84,8 | 94,7 | 94,1 | 83,9 | 95,0 | 94,4 | 78,5 | 92,9 | 91,7 |
| Sobel, $\beta$ = 80 | 83,0 | 94,5 | 93,8 | 79,5 | **95,8** | 94,9 | 73,7 | 94,2 | 92,7 |
| Sobel, $\beta$ = 100 | **81,2** | **94,4** | **93,5** | **72,9** | 94,1 | 92,5 | **70,1** | 95,1 | 93,5 |
| TanTriggs, $\beta$ = 20 | **88,4** | **96,8** | **96,5** | 84,5 | 94,7 | 94,1 | 81,6 | 91,6 | 90,7 |
| TanTriggs, $\beta$ = 40 | 88,0 | 96,4 | 96,1 | **87,9** | 95,4 | **95,0** | 82,0 | 93,2 | 92,4 |
| TanTriggs, $\beta$ = 60 | 86,9 | 95,8 | 95,4 | 87,3 | 95,0 | 94,5 | **86,3** | 94,8 | 94,4 |
| TanTriggs, $\beta$ = 80 | 85,9 | 95,8 | 95,3 | 86,4 | 94,4 | 93,9 | 83,5 | **95,5** | **94,9** |
| TanTriggs, $\beta$ = 100 | 82,3 | **94,4** | 93,7 | 77,7 | **92,9** | **91,7** | 79,0 | 91,0 | 89,8 |

TABLE 4.3 – Sensibilité (Se.), spécificité (Sp.) et précision (Pr.) moyennes (en pourcent) des SVM selon chaque configuration (type de filtre et contribution $\beta$) et de seuil d'ACP ($\tau$)

| **Filtre** | **ACP - 90%** | | | **ACP - 70%** | | | **ACP - 50%** | | |
|---|---|---|---|---|---|---|---|---|---|
| | **Se.** | **Sp.** | **Pr.** | **Se.** | **Sp.** | **Pr.** | **Se.** | **Sp.** | **Pr.** |
| Pas de filtrage | 81,0 | 94,1 | 93,2 | 76,2 | 91,9 | 90,4 | 72,3 | **84,0** | **81,9** |
| Sobel, $\beta$ = 20 | 82,5 | **94,7** | **94,0** | 78,1 | 92,2 | 90,9 | 75,6 | 87,1 | 85,4 |
| Sobel, $\beta$ = 40 | 81,3 | 93,6 | 92,7 | 81,0 | 92,4 | 91,4 | 81,3 | 89,0 | 88,1 |
| Sobel, $\beta$ = 60 | 80,6 | 93,3 | 92,4 | 80,3 | 91,4 | 90,3 | **82,2** | 89,3 | 88,5 |
| Sobel, $\beta$ = 80 | 80,0 | 92,7 | 91,6 | 79,7 | 90,8 | 89,7 | 78,6 | 89,7 | 88,4 |
| Sobel, $\beta$ = 100 | 79,2 | 92,5 | 91,4 | 79,1 | 89,9 | 88,6 | **71,3** | 92,5 | 90,5 |
| TanTriggs, $\beta$ = 20 | **83,1** | 93,9 | 93,2 | 80,0 | 92,0 | 90,9 | 78,7 | 87,5 | 86,3 |
| TanTriggs, $\beta$ = 40 | 82,8 | 93,8 | 93,1 | **81,5** | 92,7 | 91,8 | 81,0 | 91,0 | 90,0 |
| TanTriggs, $\beta$ = 60 | 81,7 | 93,1 | 92,2 | 81,0 | **92,8** | **91,9** | **82,2** | 92,9 | 92,0 |
| TanTriggs, $\beta$ = 80 | 80,2 | 92,2 | 91,2 | 78,6 | 92,4 | 91,2 | 80,6 | **93,2** | **92,2** |
| TanTriggs, $\beta$ = 100 | **74,9** | **90,9** | **89,2** | **73,7** | **87,9** | **86,0** | 76,4 | 86,6 | 85,1 |

TABLE 4.4 – Sensibilité (Se.), spécificité (Sp.) et précision (Pr.) moyennes (en pourcent) des RF selon chaque configuration de filtrage (type de filtre et contribution $\beta$) et de seuil d'ACP ($\tau$)

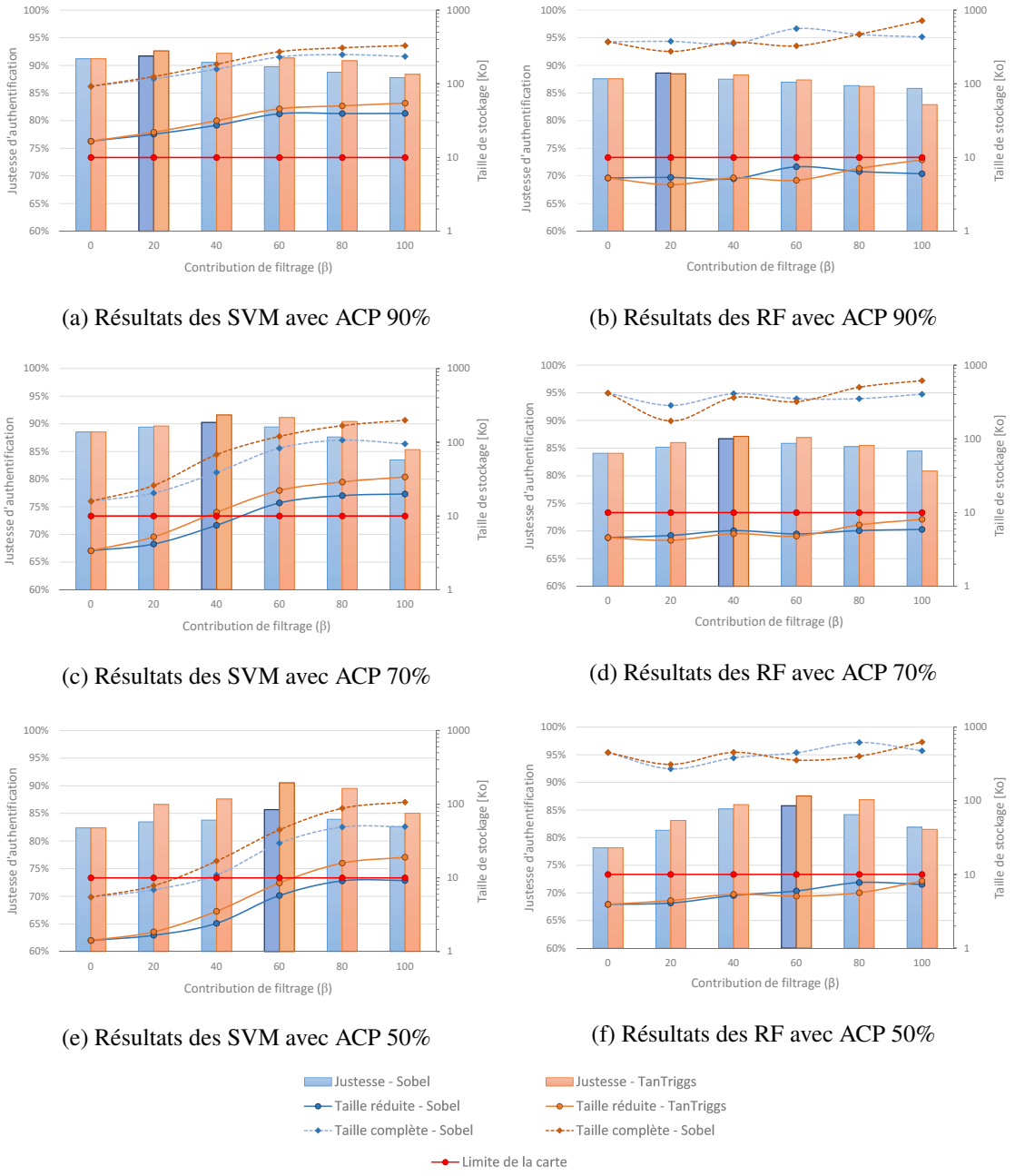
Il a été démontré, lors de nos premières expérimentations portant sur la compression de ces classifieurs [218] (comparaison des algorithmes de compression en annexe B.1), que l'algorithme "BZip2" permet de réduire le plus efficacement les SVM et les RF, et ceci de manière réversible, parmi les différentes techniques de compression sans perte étudiées ("BZip2", "LZMA", "Deflate" ou encore "PPMd" ). Les ratios de compression résultants de l'algorithme BZip2, illustrés par la table 4.5, atteignent 69,75% en moyenne sur les SVM et 97,87% pour les RF. L'écart entre ces deux valeurs démontre une plus grande difficulté de l'algorithme à compresser les informations du modèle SVM. Cette différence s'explique de par la structure des données résultant de l'apprentissage : les paramètres de classification sont stockés dans un fichier texte suivant la syntaxe "xml" mais ne traduisent pas le même type d'informations (échantillon de ces fichiers disponibles en annexe A.1.3). En effet, les SVM sont composés des coordonnées des différents vecteurs supports dont la dimension est liée à celle des données d'entrées (nombre de valeurs propres conservées) et la quantité dépend de difficulté à séparer les deux classes (répartition des caractéristiques biométriques). En comparaison, les RF sont composés d'un plus grand nombre de structures fixes (en-tête, en-queue, etc) introduisant chacune les paramètres et les nœuds de chaque arbre.

Il est possible de réduire encore la taille de stockage via une étape supplémentaire, entre l'apprentissage et la compression, consistant à modifier le nombre de décimales des paramètres sauvegardés par les classifieurs. En effet, ces paramètres possèdent une précision sur 8 décimales pour les SVM et 16 pour les RF. Une telle précision n'est pas nécessaire pour conserver une justesse d'authentification acceptable, la taille de stockage étant une de nos contraintes principales. A partir de ce constat, nous proposons de tronquer ces paramètres à seulement 4 décimales. Les résultats, illustrés par la table 4.5, montrent qu'une telle réduction de la précision décimale permet de réduire la taille de stockage initiale (avant compression) de 30,88% pour les SVM et de seulement 2,54% pour les RF. Cette différence s'explique une nouvelle fois par la structure des données de ces modèles de classification. Ces chiffres traduisent bien une réduction des données mais à un taux plus faible que la compression seule. En revanche, la combinaison de cette Réduction de Précision de Paramètres (RPP) avec la compression BZip2 réduit encore plus efficacement la taille finale du classifieur, jusqu'à 37% supplémentaires en moyenne en comparaison avec la taille résultant de la compression BZip2 seule (e.g taille originale : 617Ko ; Taille après compression : 13Ko [-97,9% par rapport à la taille originale] ; RPP+BZip2 : 8Ko [-36,5% par rapport à la taille après la compression seule]). En d'autres termes, la compression appliquée après la RPP donne, par rapport à la taille originale, des taux de réduction de 81,12% pour les SVM et de 98,57% pour les RF.

| Classifieur | RPP | BZip2 seule | RPP+BZip2 |
|---|---|---|---|
| RF | 2,54 % | 97,87 % | 98,57 % |
| SVM | 30,88 % | 69,75 % | 81,12 % |

TABLE 4.5 – Ratio de réduction de la taille de stockage des modèles de classification par rapport à leur taille originale selon différentes techniques

Considérant toutes les configurations de l'étude et les tailles de stockage présentes sur les graphes de la figure 4.11, les tailles de stockages extrêmes des SVM décroissent donc de [2 Ko ; 100 Ko] avec la compression seule (non représentée) à [1 Ko ; 66 Ko] avec la RPP+Bzip2, tandis que les RF diminuent de [5 Ko ; 16 Ko] à [3 Ko ; 11 Ko], ce qui représente un gain conséquent pour notre application. De plus, contre toute attente, les tests réalisés sur les classifieurs résultants nous ont montré que ce processus complet de réduction de taille de stockage n'a aucun impact sur la justesse d'authentification. Par conséquent il peut être envisageable de réduire d'avantage la précision décimale, mais cela n'a pas été considéré par anticipation d'une éventuelle instabilité, la réduction actuelle étant jugée suffisante.




(a) Résultats des SVM avec ACP 90%

(b) Résultats des RF avec ACP 90%

(c) Résultats des SVM avec ACP 70%

(d) Résultats des RF avec ACP 70%

(e) Résultats des SVM avec ACP 50%

(f) Résultats des RF avec ACP 50%

FIGURE 4.11 – Justesse et tailles de stockage des SVM et RF selon les seuils d'ACP et le filtrage

La justesse d'authentification est similaire entre nos deux classifieurs. Cependant, on peut observer que les SVM donnent de meilleurs résultats avec une justesse maximale d'environ 93% contre environ 89% pour les RF. La taille de stockage des modèles est à la fois impactée par leur structure de données (affecte les performances de compression) et par le nombre de valeurs propres conservées. La diminution de leur nombre réduit non seulement la taille de stockage des SVM, mais également légèrement leur justesse. En comparaison, ce seuillage d'ACP augmente très légèrement la taille de stockage des RF tandis que leur justesse diminue de la même manière que les SVM. De plus, pour chaque nouvelle réduction du nombre de valeurs propres, on remarque que le maximum de justesse est obtenu pour une plus grande valeur de $\beta$ (quel que soit le filtrage). Il y a donc un compromis à considérer quant au nombre de valeurs propres à conserver afin de maintenir des performances d'authentification optimales.

Pour les faibles valeurs de $\beta$, quand le nombre de valeurs propres diminue, il n'y a plus assez de caractéristiques de visages pour distinguer les deux classes de manière idéale. Par conséquent, les valeurs optimales de justesse d'authentification sont obtenues pour des valeurs supérieures de $\beta$, induisant une augmentation du nombre de valeurs propres conservées comme le montre la table 4.2. En revanche, il est important de noter que cette diminution n'excède pas 10% de justesse.

La table 4.6 résume les différentes configurations répondant à nos contraintes de tailles de stockage (modèle < 8 Ko), pour lesquelles les modèles donnent lieu à des résultats que nous considérons acceptables, c'est à dire avec des valeurs de sensibilité, spécificité, précision et justesse supérieures à 80%. Sur les 66 configurations évaluées, seules 23 correspondent à nos contraintes. Les valeurs extrêmes de chaque classifieurs sont colorées de la même manière que les tables précédentes. Il est notable que les valeurs voisines de ces extrémums sont relativement proches. On peut donc observer un gain conséquent en taille de stockage pour une légère réduction des performances.

| **Classifieur** | **ACP** | **Filtre** | **Se. [%]** | **Sp. [%]** | **Pr. [%]** | **Ju. [%]** | **Taille [Ko]** |
|---|---|---|---|---|---|---|---|
| SVM | 70 % | Pas de filtrage | 83,9 | 93,1 | 92,4 | 88,5 | 3,4 |
| SVM | 70 % | Sobel, $\beta = 20$ | 84,9 | 94,0 | 93,4 | 89,4 | 4,2 |
| SVM | 70 % | Sobel, $\beta = 40$ | **86,0** | 94,5 | 94,0 | **90,3** | **7,5** |
| SVM | 70 % | TanT, $\beta = 20$ | 84,5 | **94,7** | **94,1** | 89,6 | 5,2 |
| SVM | 50 % | TanT, $\beta = 20$ | 81,6 | 91,6 | 90,7 | 86,6 | **1,9** |
| SVM | 50 % | TanT, $\beta = 40$ | 82,0 | 93,2 | 92,4 | 87,6 | 3,5 |
| RF | 90 % | Pas de filtrage | 81,0 | 94,1 | 93,2 | 87,6 | 5,3 |
| RF | 90 % | Sobel, $\beta = 20$ | 82,5 | **94,7** | 94,0 | 88,6 | 5,4 |
| RF | 90 % | Sobel, $\beta = 40$ | 81,4 | 93,6 | 92,7 | 87,5 | 5,1 |
| RF | 90 % | Sobel, $\beta = 60$ | 80,6 | 93,3 | 92,4 | 86,9 | 7,5 |
| RF | 90 % | TanT, $\beta = 20$ | 83,1 | 93,9 | 93,1 | 88,5 | 4,3 |
| RF | 90 % | TanT, $\beta = 40$ | 82,8 | 93,8 | 93,0 | 88,3 | 5,3 |
| RF | 90 % | TanT, $\beta = 60$ | 81,7 | 93,1 | 92,2 | 87,4 | 4,9 |
| RF | 90 % | TanT, $\beta = 80$ | **80,2** | 92,2 | 91,2 | 86,2 | 7,1 |
| RF | 70 % | Sobel, $\beta = 40$ | 81,0 | 92,4 | 91,4 | 86,7 | 5,7 |
| RF | 70 % | Sobel, $\beta = 60$ | 80,3 | 91,4 | 90,3 | 85,9 | 5,1 |
| RF | 70 % | TanT, $\beta = 40$ | 81,5 | 92,7 | 91,8 | 87,1 | 5,2 |
| RF | 70 % | TanT, $\beta = 60$ | 81,0 | 92,8 | 91,9 | 86,9 | 4,8 |
| RF | 50 % | Sobel, $\beta = 40$ | 81,3 | **89,0** | **88,1** | **85,2** | 5,2 |
| RF | 50 % | Sobel, $\beta = 60$ | 82,2 | 89,3 | 88,5 | 85,8 | 6,0 |
| RF | 50 % | TanT, $\beta = 40$ | 81,0 | 91,0 | 90,0 | 86,0 | 5,4 |
| RF | 50 % | TanT, $\beta = 60$ | 82,2 | 92,9 | 92,0 | 87,5 | 5,1 |
| RF | 50 % | TanT, $\beta = 80$ | 80,6 | 93,2 | 92,2 | 86,9 | 5,7 |

TABLE 4.6 – Résultats des modèles de ML classique correspondant à nos contraintes

## 4.3/ Apprentissage profond : réseaux ultra-légers

De par sa place grandissante dans l'état de l'art et l'intérêt qu'il suscite, nous avons choisi de comparer les travaux précédents au *"Deep Learning"* et plus précisément aux réseaux de convolutions (CNN - *"Convolutional Neural Network"*). Plus récents que les méthodes précédentes, plus efficaces mais également plus gourmands en ressources, ils nécessitent une capacité de stockage largement supérieure aux méthodes classiques (dizaine voire centaine de mégaoctets). Contrairement à ces dernières, les CNN intègrent directement une partie détection et extraction de caractéristiques de par leurs nombreuses couches cachées *("hidden layers")* mettant en œuvre des convolutions en cascade. Cette approche particulièrement efficace pour un grand nombre d'applications présente cependant quelques difficultés dans sa mise en œuvre. En particulier la phase d'apprentissage qui apparait comme particulièrement critique dans sa préparation et sa réalisation. En effet, elle requiert une quantité très importante d'images annotées pour généraliser l'aspect visuel d'une classe (millions d'images), dont l'obtention ou la réalisation n'est pas toujours aisée en fonction des applications. De plus, les ressources matérielles conséquentes ainsi que la durée nécessaire à la convergence de la phase d'apprentissage (dizaines voire milliers d'heures de calculs) peuvent également représenter un frein à la mise en œuvre des CNNs.
Nous avons alors considéré le *"Transfert Learning"* [219] qui représente dans l'état de l'art une alternative intéressante à cette phase d'apprentissage "classique" des CNN. Cette technique permet d'utiliser un réseau CNN pré-entraîné dans un contexte donné et de le spécialiser dans une nouvelle tâche spécifique en effectuant un nouvel entraînement de ses dernières couches pour de nouvelles classes, après avoir "gelé" ses couches cachées (*"frozen hidden layers"*, utilisées en tant que détecteurs de caractéristiques). L'apprentissage nécessite par conséquent beaucoup moins d'images et de temps. Une étude complémentaire et similaire au Machine Learning classique est donc proposée, avec pour but, en appliquant le *"Transfer Learning"* sur des CNN pré-entraînés très légers, de confronter les aptitudes du *"Deep Learning"* à celles des méthodes exploitées précédemment et d'observer d'éventuelles limitations du *"Transfer Learning"*. Nous considérons des réseaux comme étant légers lorsque leur nombre de paramètres (donnant une indication sur la taille de stockage nécessaire au réseau) est inférieur à $5.10^6$ et la quantité d'opérations (*"MACs" - "Memory Access Cost" ou "Multicply and ACcumulates"*, indiquant la vitesse de fonctionnement du réseau) est inférieure à $300 \times 10^6$. Cette étude met en évidence les performances intrinsèques de cette méthode en terme de qualité de décision mais aussi en considérant la taille du modèle à stocker pour respecter les contraintes liées à notre application.

### 4.3.1/ Mise en œuvre

#### 4.3.1.1/ Transfer Learning et réseaux "légers" dans la littérature

Le transfer learning permettant de ré-entraîner un réseau tout en conservant ses aptitudes de détection, il est nécessaire de sélectionner soigneusement le réseau de base. En effet, afin de correspondre à nos contraintes, le réseau doit être léger (peu de paramètres et d'opérations) mais présenter l'avantage d'avoir appris à détecter des caractéristiques soit très proches de nos nouvelles données, soit extrêmement variées afin d'être en capacité de construire une carte de caractéristiques au plus proche de nos besoins. Le nombre d'objets connectés grandissant, l'intégration d'intelligence embarquée est devenue fondamentale, motivant donc la recherche dans cette direction. De nouveaux réseaux toujours plus légers voient alors le jour et certains correspondent potentiellement à nos attentes, comme "MobileNets", "SqueezeNet" [220], "Shufflenet" [221], "PeleeNet" [222], ou encore "EffNet" [223], tous entraînés à partir du challenge "ImageNet" [224]. Nous proposons ici de ré-entraîner les CNN "MobileNets" dans leur version 1 [225] et version 2 [226]. Ces réseaux sont dédiés à une utilisation embarquée sur smartphones. Entraînés à l'origine à la reconnaissance d'objets très divers, aux travers de 1000 classes à partir de millions d'images pour le challenge

ImageNet, leurs couches cachées sont capables de détecter des caractéristiques très variées et donc ces réseaux sont facilement réutilisables pour des tâches très différentes.

La particularité de MobileNet est de remplacer les convolutions conventionnelles par de nouvelles appelées *"Depthwise Separable Convolutions"*. Une convolution standard filtre les données d'entrées via une fenêtre glissante sur tous les canaux simultanément et combine les résultats en une nouvelle série de données. Avec un but identique, les convolutions *"depthwise separable"* divisent ces étapes en deux couches distinctes : une convolution appelée *"depthwise"* suivie d'une convolution 1x1 appelée *"pointwise"*. Cette factorisation permet de grandement réduire le nombre de calculs et la taille du modèle. L'annexe A.2.2 présente de plus amples précisions sur ces convolutions, ainsi que sur les architectures des réseaux MobileNet dans leur versions 1 et 2.
Soit $M$ le nombre de canaux d'entrée d'une couche de convolution (*"input depth"*, 3 pour une image couleur, dépend de la couche précédente pour une *"feature map"*), $N$ le nombre de filtres (et donc nombre de canaux de sortie - *"ouput depth"*) et $D_K$ la taille du noyau de convolution. En considérant un pas de un pixel, le nombre de multiplications d'une convolution *"depthwise separable"* est définit par $[D_F.D_F.M.(N + D_K.D_K)]$, et $[D_F.D_F.M.N.D_K.D_K]$ définit le nombre de multiplications pour une convolution standard, où $D_F$ correspond à la taille de l'image traitée. On observe bien que les convolutions *"depthwise separable"* sont moins coûteuses en ressources. Les réseaux MobileNet utilisent des convolutions *"depthwise"* de taille $D_K = 3$.

Bien que les deux versions de base de MobileNet soient déjà très légères, l'utilisation grandissante de l'intelligence artificielle embarquée, comme dans nos smartphones, requiert des modèles toujours plus légers et rapides. Afin de construire à partir de MobileNet ces réseaux plus légers et moins gourmands en puissance de calcul, deux paramètres, $\alpha$ et $\rho$, ont été introduits [225].
Appelé "facteur de largeur" ("width multiplieur"), le paramètre $\alpha$ permet d'affiner uniformément le réseau à chaque couche, ce qui réduit les calculs nécessaires et le nombre de paramètres par approximativement $\alpha^2$. Pour une couche et un paramètre $\alpha$ donnés, le nombre de canaux d'entrée "$M$" devient "$\alpha.M$" et le nombre de canaux de sortie "$N$" (nombre de filtres) devient "$\alpha.N$", avec $\alpha \in ]0; 1]$. Différentes valeurs de ce paramètres ont été définies : $\{1\,;\, 0{,}75\,;\, 0{,}5\,;\, 0{,}35/0{,}25$ [1]$\}$. Le réseau original correspond donc à $\alpha = 1$ et les réseaux réduits à $\alpha < 1$.
Le second paramètre, appelé facteur de résolution *("resolution multiplier")*, permet également de réguler le nombre de calculs nécessaires. Appliqué aux images d'entrée, la constitution interne de chaque couche est ensuite réduite par ce même facteur $\rho$ ($\in ]0, 1]$), ce qui a pour effet de réduire les besoins en calculs d'environ $\rho^2$. Le coût calculatoire d'une convolution *depthwise separable* (pour MobileNet v1), prenant en compte les paramètres $\alpha$ et $\rho$ est alors de $[\rho^2.D_F^2.\alpha.M.(\alpha.N + D_K^2)]$.

Différentes valeurs de $\rho$ ont été définies de manière à implicitement correspondre à certaines dimensions d'images d'entrées : $\{224\,;\, 192\,;\, 160\,;\, 128\,;\, 96$ [2]$\}$. Le réseau original correspond donc à $\rho = 1$, et les réseaux réduits à $\rho < 1$. Le tableau 4.7 récapitule brièvement les caractéristiques de ces réseaux. On y retrouve les deux versions de MobileNet ainsi que leurs différentes déclinaisons en réseaux réduits de par les variations des paramètres $\alpha$ et $\rho$. Permettant avant tout de moduler la vitesse et l'encombrement au niveau stockage selon les besoins, ces paramètres ont évidemment un impact sur les performances prédictives des réseaux.

1. 0,35 : Seulement MobileNet v2 ; 0,25 : Seulement MobileNet v1
2. Seulement pour MobileNet v2

| Réseaux disponibles | MACs | Paramètres | Justesse [%] | Taille [Mo] |
|---|---|---|---|---|
| **v1_0.25_128** | **14**.$10^6$ | **0,47**.$10^6$ | **41,5** | **2,00** |
| v1_0.25_160 | 21.$10^6$ | 0,47.$10^6$ | 45,5 | 2,00 |
| v1_0.25_192 | 34.$10^6$ | 0,47.$10^6$ | 47,7 | 2,00 |
| **v1_0.25_224** | **41**.$10^6$ | **0,47**.$10^6$ | **49,8** | **2,00** |
| v1_0.50_128 | 49.$10^6$ | 1,34.$10^6$ | 56,3 | 5,20 |
| | | ... | | |
| v1_1.0_224 | 569.$10^6$ | 4,24.$10^6$ | 70,9 | 16,70 |
| v2_0.35_96 | 11.$10^6$ | 1,66.$10^6$ | 45,5 | 6,60 |
| **v2_0.35_128** | **20**.$10^6$ | **1,66**.$10^6$ | **50,8** | **6,60** |
| v2_0.35_160 | 30.$10^6$ | 1.66.$10^6$ | 55,7 | 6,60 |
| v2_0.35_192 | 43.$10^6$ | 1.66.$10^6$ | 58,2 | 6,60 |
| **v2_0.35_224** | **59**.$10^6$ | **1,66**.$10^6$ | **60,3** | **6,60** |
| v2_0.50_96 | 32.$10^6$ | 1,96.$10^6$ | 57,7 | 7,75 |
| | | ... | | |
| v2_1.0_224 | 300.$10^6$ | 3,47.$10^6$ | 71,8 | 13,76 |

TABLE 4.7 – Caractéristiques des réseaux MobileNet v1 [227] et v2 [228] et leurs performances originales en terme de justesse de reconnaissance d'objet sur les 1000 classes du challenge ImageNet

Ces différentes variantes de réseaux disponibles sont nommées comme suit : "$\vartheta$_$\alpha$_$\delta$", où $\vartheta$ est la version du réseau, $\alpha$ le facteur de largeur et $\delta$ la dimension d'image d'entrée définie par $\rho$. Ce tableau confirme le fait que la taille de stockage dépend directement du nombre de paramètres (et donc de la profondeur du réseau) et la justesse d'authentification dépend du nombre d'opérations (lui même lié à la taille des images). Hormis la justesse, qui dépend également de la faculté du réseau à discriminer les classes, le nombre d'opérations (MACs) et le nombre de paramètres restent inchangés après la phase de ré-entraînement. Les contraintes du système sont étroitement liées à la taille du stockage et aux temps de calculs.
Le transfer learning permettant de spécialiser un réseau pré-entraîné à une tâche spécifique, les faibles performances de reconnaissance d'objets liées à ces réseaux se verront nécessairement améliorées et ne font donc pas partie de nos critères de sélection. Bien que leurs tailles de stockage demeurent excessives vis à vis des contraintes de stockage visées par notre application, les réseaux les plus légers de chaque version présentent le meilleur compromis et ont donc été choisis, avec pour chacun les tailles extrêmes d'images d'entrées : v1_0.25_(128/224), v2_0.34_(128/224). L'apprentissage original des réseaux sélectionnés donne respectivement 41,5 %, 49,8 %, 50,8 % et 60,3 % de justesse pour la reconnaissance d'objets sur les 1000 classes du challenge ImageNet. En comparaison, sur le même challenge, la justesse des réseaux connus donne : 71,5% avec 138.$10^6$ paramètres pour "VGG16" [229], 77% avec 60.$10^6$ paramètres pour "ResNet v2 101" [230], 80,2% avec 25.$10^6$ paramètres pour "Inception v4" [231], 70,9% avec $5,2.10^6$ paramètres pour "ShuffleNet" [221], ou encore 57,5% avec $1,2.10^6$ paramètres pour "SqueezeNet" [220]. L'objectif est donc, de par l'application du *"Transfer Learning"* (ou transfert de connaissances), d'augmenter cette justesse pour deux nouvelles classes non présentes dans les données d'apprentissage originales (sujet autorisé et personnes non autorisées).

#### 4.3.1.2/ Protocoles pour le transfer learning et tests d'authentification

De la même manière que précédemment, l'objectif est d'effectuer une authentification de visages, par conséquent, un réseau est ré-entraîné par sujet avec deux classes (personne habilitée, personnes non autorisées). Afin d'exposer les limitations du transfer learning, nous proposons d'entraîner les réseaux suivant deux configurations différentes : dans un premier temps avec 1000 images par classe et 40 images par classe dans un second temps (autant que pour les SVM et RF). Le transfer learning nécessite de considérer un type d'images similaire à celles utilisées par le réseau original (dimensions, nombre de canaux, etc). Par conséquent, contrairement aux méthodes classiques utilisées précédemment, les CNN utilisent des images couleurs sur 3 canaux. Aucun filtrage n'est appliqué ici, cette tâche étant réalisée par les différentes convolutions de chaque couche. En revanche, chaque image de visage a été également alignée au préalable.

Lors du processus de transfer learning, un certain nombre de paramètres d'apprentissage peuvent être ajustés afin d'optimiser les performances de classification selon les images d'entrée. Ces paramètres couvrent entre autres le nombre de boucles d'entrainement (ou itérations d'apprentissage - $T_{steps}$), le taux d'apprentissage (*"learning rate"* - LR), la taille des "lots" d'images (*"batch"*) dédiés à l'apprentissage, à l'évaluation de chaque boucle et à la validation globale. Une étude préliminaire a été réalisée sur ces paramètres et les premiers résultats nous ont permis de fixer la taille du lot d'apprentissage à 32 images. Afin de garantir la convergence des réseaux, nous avons établi les relations suivantes entre le LR et le nombre de boucles d'entraînement :

$$\begin{cases} LR & = 10^{-\sigma} \\ T_{steps} & = 1000.2^{\sigma} \end{cases} \tag{4.7}$$

La figure 4.12 donne la sensibilité, la spécificité et la justesse en fonction des différentes valeurs du learning rate étudiées. Ces valeurs ont été calculées en faisant diminuer graduellement notre paramètre $\sigma$ de 5 à 1. En suivant la relation 4.7 établie précédemment, le nombre d'itération d'apprentissage correspond respectivement à $\{32000; 16000; 8000; 4000; 2000\}$. Ce graphique nous montre que les meilleures performances sont données par un learning rate de 0,0001 ($\sigma = 4$). Néanmoins, un compromis doit être trouvé quant à la quantité de calculs. Les performances du réseau associé au learning rate de 0,001 semblent très proches de ces performances maximales. La suite de ce manuscrit ne traitera que de la combinaison qui nous a semblé optimale : nous avons donc fixé le paramètre $\sigma$ à 3, donnant un learning rate de $0,001$ et 8000 itérations d'apprentissage.

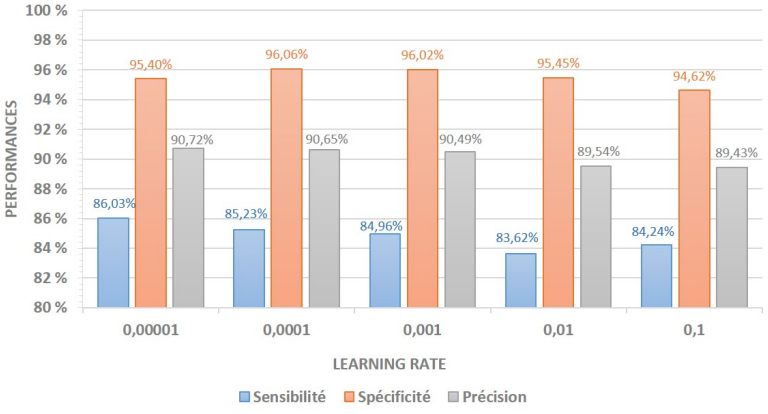



Figure 4.12 – Performances moyennes de MobileNet v1_0.25_128 selon le Learning Rate

Les réseaux résultants des apprentissages par transfert de connaissances sont ensuite soumis à un processus de "quantification" (*"Quantization process"*). Cette quantification (ici de type *"rounded weights"*) conserve les poids des paramètres du réseau en tant que données au format flottant, mais les arrondit de telle sorte que le réseau résultant conserve sa taille de stockage tout en contenant beaucoup plus de séquence d'octets répétitifs. Ainsi, une compression additionnelle du réseau permet de grandement réduire sa taille de stockage. Contrairement aux SVM et RF, le réseau sauvegarde ses poids sous forme de "protocole buffer" [232] (mécanisme de Google sérialisant la structure de données), l'étape de quantification est alors indispensable pour compresser le réseau, à défaut, un algorithme de compression atteindrait un taux inférieur à 10%. Une étude préliminaire nous a permis d'observer que, dans le cas du protocole buffer, l'algorithme de compression LZMA était le plus approprié (cf annexe B.1), que ce soit avec ou sans cette étape de quantification. Les taux de compression résultants, atteignant environ 76%, sont détaillés dans le tableau 4.8.

| Network | Quant. | LZMA seule | Quant. + LZMA |
|---|---|---|---|
| MobileNet v1_025 | 0% | 8,75% | 76,27% |
| MobileNet v2_035 | 0% | 8,24% | 75,75% |

TABLE 4.8 – Taux de réduction de la taille des réseaux MobileNet, par rapport à leur taille originale, après la phase de quantificiation (Quant.), la compression LZMA et la quantification suivie de la compression LZMA

De la même manière que précédemment, dans une démarche de simplification de la compréhension des différentes phases de recherche, nous proposons de résumer les étapes de l'étude sous forme schémas bloc.

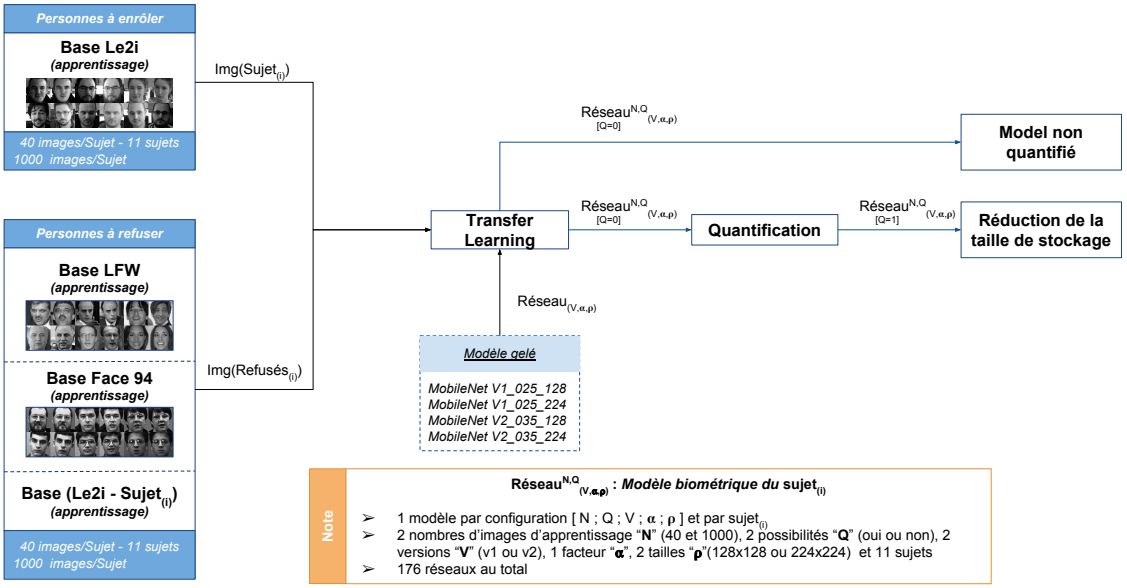



FIGURE 4.13 – Protocole d'apprentissage des CNN par transfert de connaissances

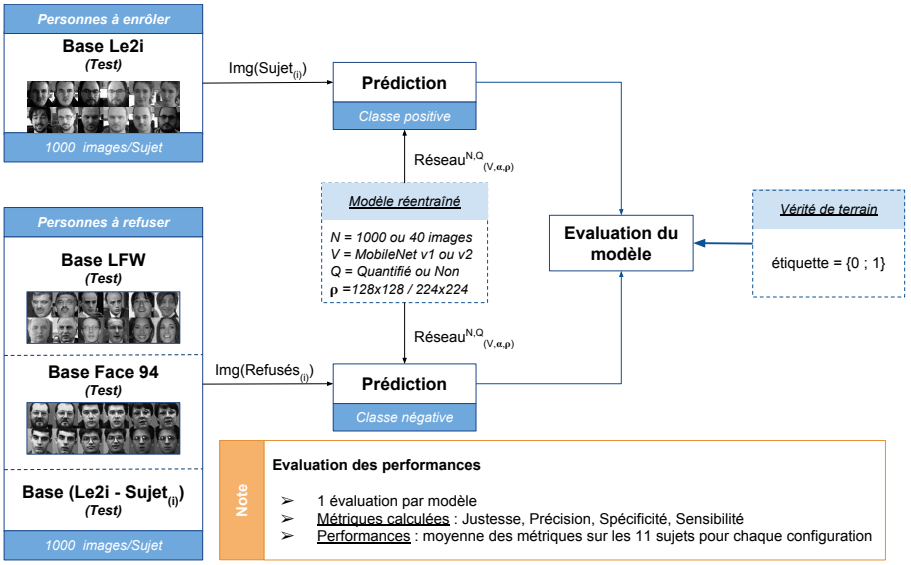



FIGURE 4.14 – Protocole de test et d'évaluation des CNN ré-entraînés

### 4.3.2/ RÉSULTATS DE L'ÉTUDE

Les résultats des méthodes du transfer learning représentent les comportements des CNN ré-entraînés, pour chacune des versions de MobileNet choisies. La table 4.9 présente la sensibilité (Se.), la spécificité (Sp.) et la précision (Pr.) de ces réseaux selon le nombre d'images par classe lors de l'apprentissage, leur dimension et selon le fait que le réseau ai été quantifié ou non. De la même manière que pour le Machine Learning classique précédemment, nous n'imposons pas de seuil sur le score de la prédiction.

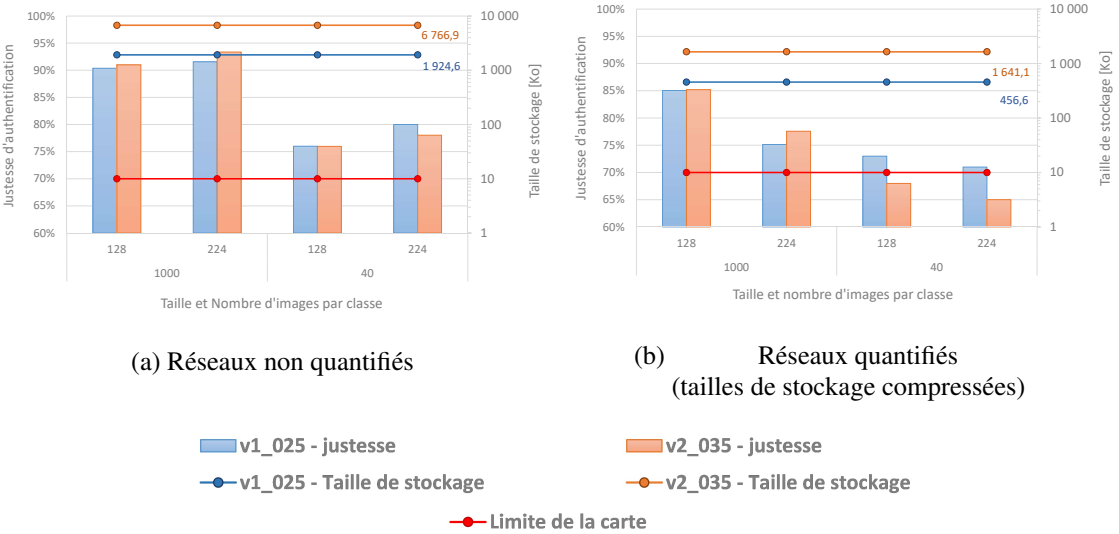
| Réseau | | 1000 images par classe | | | 40 images par classe | | |
|---|---|---|---|---|---|---|---|
| | | Se. | Sp. | Pr. | Se. | Sp. | Pr. |
| **v1_025_128** | **Non Quantifié** | **86,0** | **94,8** | **94,1** | **60,7** | 88,6 | 83,1 |
| **v1_025_224** | | 87,2 | 95,9 | 95,3 | **76,1** | **85,8** | **79,2** |
| **v2_035_128** | **Non Quantifié** | 86,2 | 96,8 | 96,2 | 62,9 | 89,1 | 85,2 |
| **v2_035_224** | | **89,6** | **97,2** | **96,9** | 76,0 | **90,1** | **86,8** |
| **v1_025_128** | **Quantifié** | **79,5** | 90,6 | 90,3 | 52,1 | 85,5 | **81,3** |
| **v1_025_224** | | **62,2** | **88,1** | **89,7** | **54,7** | **83,4** | 75,0 |
| **v2_035_128** | **Quantifié** | 77,0 | **93,5** | **93,3** | 46,0 | 86,7 | 80,8 |
| **v2_035_224** | | 63,5 | 91,7 | 92,3 | **31,9** | **88,4** | **63,7** |

TABLE 4.9 – Sensibilité (Se.), spécificité (Sp.) et précision (Pr.) moyennes, en pourcent, des réseaux MobileNet selon chaque configuration (version du réseau, dimensions des images et nombre d'image d'apprentissage)

Sans quantification des réseaux, ce tableau nous apprend qu'avec 1000 images par classes, les performances sont similaires à celles du Machine Learning classique pour leurs meilleures configurations. L'usage du transfert de connaissance rend donc possible une spécialisation d'un CNN avec très peu d'images, relativement à la quantité conventionnellement attribuée aux les CNNs. Cependant, l'utilisation de seulement 40 images par classe délivre de moins bonnes performances. On observe en effet une diminution inférieure à dix points pour la spécificité et jusqu'à vingt points pour la sensibilité et la précision. Au vu des valeurs de spécificité, on peut affirmer que, comme pour le Machine Learning classique, nos modèles issus du Deep Learning ont de bonnes capacités à détecter les intrusions.
Une certaine similitude est à noter en comparant les valeurs des métriques d'authentification résultant des deux versions de MobileNet sélectionnées. On observe cependant un léger avantage pour la version 2 de MobileNet, de 1% à 5%. Pour les deux quantités d'images d'apprentissages étudiées, l'influence d'une taille d'images plus grande ($224 \times 224$ contre $128 \times 128$) se distingue par une légère amélioration des performances pour chaque version de MobileNet.

La quantification post-apprentissage des réseaux induit une diminution des résultats sur ces trois métriques d'évaluation. Une fois quantifiés, les réseaux dénotent également un comportement différent face aux variations du nombre d'images pour l'apprentissage et de leur taille. En effet, la diminution de la quantité d'images par classe diminue une nouvelle fois la sensibilité, jusqu'à une valeur de 50% environ pour les plus faibles valeurs, conservant toutefois une spécificité et une précision supérieures à 80% en moyenne. Cela traduit une faible capacité à reconnaitre les personnes autorisées mais une bonne capacité à refuser les intrus. Contrairement aux réseaux non quantifiés, l'augmentation de taille des images diminue les performances de quelques points, et jusqu'à 17% pour le cas extrême. Sauf certaines exceptions, la version 2 de MobileNet améliore légèrement les résultats. On observe cependant des sensibilités très faibles avec 40 images pas classe.

Les figures 4.15a et 4.15b représentent pour chacune des versions de MobileNet choisies, respectivement avant et après leur quantification, l'influence du nombre d'images d'apprentissage et de taille de ces images sur leur justesse, mise en corrélation avec la taille du modèle.




(a) Réseaux non quantifiés

(b) Réseaux quantifiés (tailles de stockage compressées)

FIGURE 4.15 – Justesse d'authentification et taille des réseaux MobileNet avant et après quantification, selon le nombre d'images par classe et leur taille

Sur ces graphiques, l'impact du nombre d'images d'entrée sur les performances d'authentification des réseaux est remarquable. En effet, utiliser seulement 40 images par classe fait diminuer de manière significative la justesse d'authentification des deux réseaux, quelle que soit la taille de ces images. Les réseaux ré-entraînés sans quantification produisent de meilleurs résultats pour les images de plus grandes dimensions spatiales, mais une tendance inverse est à noter pour les réseaux après quantification. La justesse d'authentification de ces réseaux non quantifiés est similaire à celles des SVM/RF, mais leurs tailles de stockages sont bien plus grandes.
Pour 1000 images par classe de taille $224 \times 224$, le réseau mobileNet v2 non quantifié atteint 93% de justesse pour 6767 Ko, et la version 1 du réseau atteint 92% pour 1925 Ko. En comparaison, pour une justesse proche, les SVM ont une taille de stockage compressée de 11 Ko (68 Ko non compressés) pour une justesse de 92% (ACP à 70% et filtrage TanTriggs à 40%) et les RF ont une taille de stockage compressée de 5,3Ko (376 Ko non compressée) avec une justesse de 89% (ACP à 90% et filtrage de Sobel à 20%). La quantification, nécessaire à la compression, mène à une diminution beaucoup trop importante de la justesse et des autres métriques d'évaluation. De plus, les tailles de stockage des réseaux après compression ne sont pas suffisamment réduites pour répondre à nos contraintes.

Une autre solution permettant de réduire la taille de stockage des réseaux consiste à extraire les couches de classification. En effet, en ne considérant aucune quantification, les couches cachées étant gelées et utilisées comme extracteurs de caractéristiques, elles restent identiques quel que soit le sujet. À partir de ce constat, de la même manière que pour les espaces de visages issu de l'ACP, sauvegarder ces couches cachées localement et les dernières couches sur la carte sans contact représente une piste envisageable pour réduire la quantité de mémoire.
Afin de ne représenter que 8 Ko de données, considérant la taille complète des réseaux (1925 Ko pour Mobilenet v1 et 6767 Ko pour Mobilenet v2), le nombre de paramètres de ces dernières couches de classification ne doit pas représenter plus 0,42% du nombre de paramètres total pour MobileNet v1 (ou 1,7% pour le réseau quantifié), et 0,12% pour MobileNet v2 (ou 0,49% pour le réseau quantifié). Hors, d'après Howard et.al [225], tableau 4.10, les dernières couches de classification réseau complet (MobileNet v1_1.0_224) représentent 24,33% du nombre de paramètres total, ce qui correspond à 468 Ko. Sur le même principe et d'après la documentation du framework Deep Learning "Keras" [3], les couches de classification du modèle MobileNet v1_025 représentent 60% des paramètres du réseau et les couches de classification du modèle MobileNet v2_035 représentent 76,5% des paramètres du réseau. Par conséquent, cette méthode de réduction ne pourra pas permettre d'obtenir moins de 8 Ko de données biométriques par personne.

| Type | MACs | Paramètres |
|---|---|---|
| Conv 1x1 | 94,86 % | 74,59 % |
| Conv dw 3x3 | 3,06 % | 1,06 % |
| Conv 3x3 | 1,19 % | 0,02 % |
| Fully Connected | 0,18 % | 24,33 % |

TABLE 4.10 – Ressources occupées par couche pour MobileNet v1_1.0_224 [225]

3. Documentation GitHub de Keras : "Keras-Team/Keras Applications"

## 4.4/ Analyse des résultats

Dans la littérature, pour la reconnaissance ou l'authentification de visages, les CNN sont utilisés soit en tant que détecteur de caractéristiques et classifieur, soit en tant que seul extracteur de caractéristiques (de la même manière que l'ACP), la phase de classification étant assurée par une méthode additionnelle (SVM, distance euclidienne, modèle Bayesien ...). Le but de ce mode d'utilisation est de créer un modèle optimisant la répartition des données d'apprentissage, séparant au mieux les classes et minimisant la variance interclasse. Ainsi, en utilisant un classifieur très simple, les caractéristiques de nouvelles images de visages sont générées par le réseau et une classification rapide peut être effectuée. Ces CNNs, bien plus robustes aux diverses perturbations d'acquisition, nécessitent néanmoins toujours des millions d'images ainsi que des centaines voir des milliers d'heure d'apprentissage. L'objectif de ces réseaux de convolutions est d'atteindre des performances de reconnaissances toujours plus élevées (quel que soit le coût de calcul), tandis que notre objectif consiste à réduire la taille de stockage des réseaux tout en limitant au mieux la perte de performances d'authentification (afin d'être embarqué sur une unité de calcul à puissance réduite). Le tableau 4.11 met en évidence les performances actuelles de l'état de l'art.

| Réseau | Nb images | Classification | justesse |
|---|---|---|---|
| **DeepID3** [233] | $290.10^3$ | "Joint Bayesian" | 99,53 % |
| **FaceNet** [166] | $200.10^6$ | Distance Euclidienne L2 | 99,60 % |
| **VGGFace** [234] | $2,6.10^6$ | Distance Euclidienne L2 | 99,63 % |
| **Baidu** [235] | $1,2.10^6$ | "Triplet loss" | 99,70 % |

Table 4.11 – Résultats des CNN de l'état de l'art sur la reconnaissance de visages

Nous avons montré que, malgré des performances d'authentification de visages plus faibles que les taux de reconnaissances des CNNs de l'état de l'art, le Transfer Learning, appliqué à des réseaux légers, est une méthode d'apprentissage très efficace. Il permet à un réseau, dédié à une certaine tâche, de converger vers de hautes performances de classification pour une nouvelle tâche plus spécifique. En effet, cette méthode nous a permis de gagner un facteur légèrement supérieur à deux sur la justesse du réseau MobileNet le plus léger, en le spécialisant pour un problème très différent de son apprentissage original. En d'autres termes, la justesse de reconnaissance d'objets de ce réseau, entraîné à partir de millions d'images du challenge ImageNet pour 1000 classes, est de 41,5%, et la justesse d'authentification sur nos données avec seulement 2000 images d'apprentissage pour 2 classes, après le Transfer Learning, atteint 90% de justesse.
L'amélioration est cependant moins conséquente pour 40 images par classe, ce qui semble représenter une limite au transfer learning, atteignant uniquement 75% de justesse. Malgré ces résultats satisfaisant, la taille de stockage des modèles est très élevée. Les outils de réduction de données ne permettent pas de réduire suffisamment cette taille pour répondre à nos contraintes, réduisant même la justesse d'authentification.
Cette conclusion n'est pas immuable. Elle ne dépend en effet que de nos contraintes, lesquelles sont définies selon les capacités de stockage actuelles des supports sans contacts telles que les cartes RFID. Celles-ci sont amenées à évoluer dans les prochaines années, au même titre que la puissance de calculs des systèmes embarqués et que les performances générales des CNN. De plus, avec le développement massif de l'internet des objets, les travaux actuels dans le domaine du Deep Learning portent de plus en plus sur la réduction de la taille des CNN (réduction du nombre de paramètres) tout en conservant les performances qui ont fait leur notoriété.

En comparaison, le Machine Learning classique répond à nos contraintes dans certaines configurations. Les performances d'authentification des SVM et RF retenus (sur critère de taille inférieure à 8 Ko et de métriques d'évaluation supérieures à 80%) sont contenues dans l'intervalle ]80% ;95%[ et peuvent donc être considérées comme satisfaisantes. La taille de stockage de ces classifieurs en sortie directe de la phase d'apprentissage est trop élevée pour répondre à nos contraintes mais la phase de réduction des données (RPP+compression) s'avère être très efficace (réduction de 98,6% pour les RF et de 81,1% pour les SVM).
Considérant la même quantité d'images d'entrée (40 par classe), les méthodes classiques prennent l'avantage sur les CNNs en ce qui concerne les performances d'authentification. Cependant, en considérant 1000 images par classes, elles sont approximativement équivalentes. En revanche, la taille de stockage des modèles CNN se trouve être nettement supérieure aux modèles issus du Machine Learning traditionnel.

Nos contraintes étant également liées à la vitesse des traitements, une dernière étude est menée afin d'obtenir les temps de calculs et ainsi pouvoir mettre en évidence la ou les solutions les plus appropriées quant à notre application. Les méthodes exploitées précédemment sont donc testées vis à vis des conditions réelles sur une plateforme *"low cost"*. Diverses mesures sont donc effectuées sur les phases de l'apprentissage et de l'authentification et les résultats de cette étude sont exposés dans le chapitre 6 de ce manuscrit.

# 5

# Modalités du doigt : Veines et empreintes digitales

L'authentification de visages exposée précédemment ne permet pas garantir un niveau de sécurité maximal face aux usurpations d'identités. L'utilisation d'une caméra 2D ne donne en effet que des informations de cette même dimension. Il est alors possible de tromper le système en utilisant un masque ou une photo. Plusieurs solutions existent, comme l'utilisation d'une caméra de profondeur permettant d'obtenir des caractéristiques en 3 dimensions, ou la mesure complémentaire d'autres modalités biométriques.

C'est cette dernière possibilité qui a été retenue dans ce chapitre, traitant donc de l'étude de deux modalités acquises sur le doigt du sujet. Une première dite "avec traces" et une seconde "sans traces", respectivement les empreintes digitales et le réseaux vasculaire du doigt. L'emploi de données sous cutanées nécessite une acquisition particulière et réduit par conséquent les risques de fraudes. Ce chapitre présente la chaîne de traitements de ces deux modalités, les caractéristiques extraites, ainsi que la phase d'authentification. Cette authentification est effectuée par une prédiction de classe fournie par un classifieur SVM, séparant en deux catégories (acceptés ou refusés) des attributs issus de la mise en correspondance des caractéristiques extraites de deux images (caractéristiques de référence et caractéristiques de l'image courante).

## Sommaire

## 5.1/ Présentation des modalités du doigt

### 5.1.1/ Empreintes digitales

Les empreintes digitales se trouvent aux extrémités des doigts, sur la face palmaire de la dernière phalange. Elles sont formées de lignes courbes en reliefs sur la peau, aussi appelées "crêtes". Ces empreintes sont différentes sur chaque doigt et uniques pour chaque individu. Les informations fournies par les empreintes digitales, permettant d'authentifier ou d'identifier un individu, se distinguent au travers de trois niveaux de précision tel que le montre la figure 5.1.

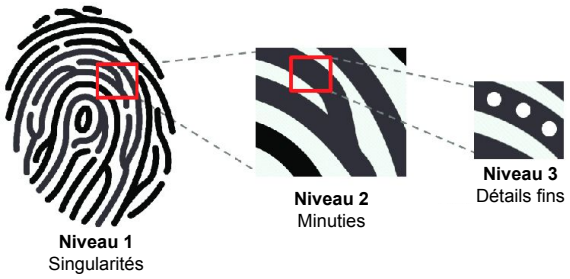



Figure 5.1 – Les niveaux de caractéristiques fournis par les empreintes digitales [236]

**Niveau 1 :**
Le niveau 1 est le plus haut niveau d'abstraction de caractéristiques. Il se compose de formes générales, aussi appelées singularités. On distingue trois grands groupes : les boucles, les verticilles et les arches. Chacun de ces groupes possède diverses variations comme le montre la figure 5.2. Ces singularités sont approximées par l'estimation de l'orientation locale des crêtes et peuvent être détectées à partir d'images basses résolution.

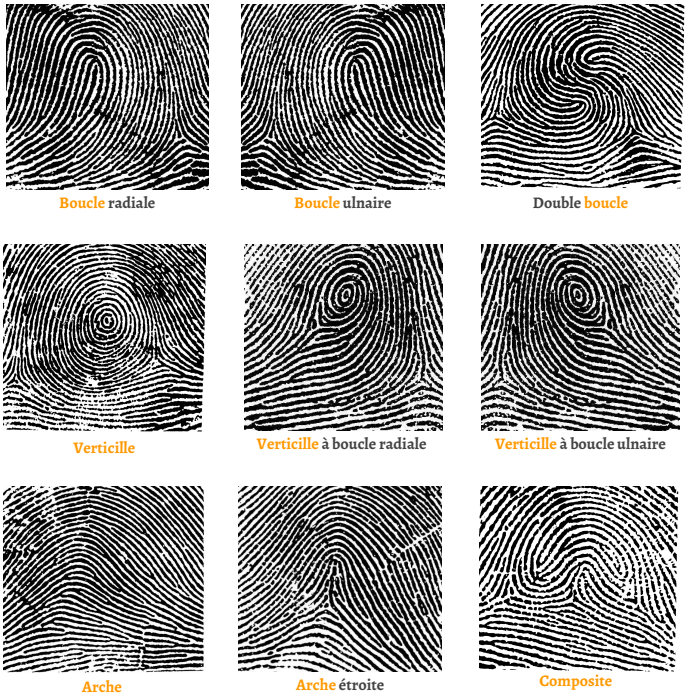



Figure 5.2 – Empreintes digitales : caractéristiques de niveau 1 - Principales singularités

**Niveau 2 :**
Le niveau 2 des caractéristiques regroupe des informations plus précises sur ces crêtes, illustrées par la figure 5.2. Ce niveau de représentation est principalement composé de données précises telles que les débuts et les fins de lignes (points rouges), ainsi que les bifurcations et les croisements de lignes (points verts). Comme le montre cette figure, il comporte également des motifs formés de ces données et des positions relatives des crêtes par rapport aux voisines. Ces motifs, appelés minuties, nécessitent des images de meilleure qualité et de plus haute résolution.

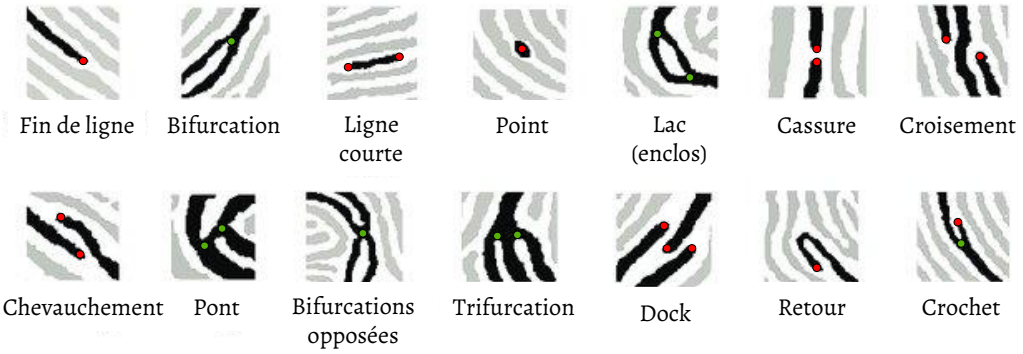



FIGURE 5.3 – Empreintes digitales : caractéristiques de niveau 2 - Principales minuties [236]

**Niveau 3 :**
Le niveau 3 constitue le plus bas niveau de représentation des caractéristiques des empreintes digitales. Il comporte des données beaucoup plus précises et son exploitation nécessite des images de très haute résolution et de haute qualité. Comme le montre la figure 5.4, on distingue les pores de la peau, la forme précise du contour des crêtes, et d'autres données précises propres à la peau.

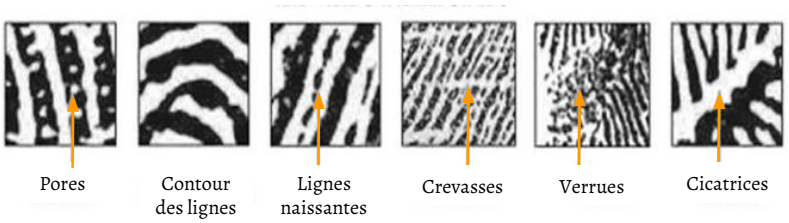



FIGURE 5.4 – Empreintes digitales : caractéristiques de niveau 3 - Principaux détails fins [236]

De la même manière que pour les visages, l'acquisition des empreintes digitales est soumise à diverses perturbations pouvant troubler l'extraction de caractéristiques et donc la correspondance. La plupart des systèmes d'acquisition de cette modalité étant soit des systèmes avec contact, soit des systèmes "fermés", les problèmes liés au contrôle de l'éclairage ne s'appliquent pas (ou peu). Les empreintes digitales sont alors plutôt soumises à des perturbations affectant la surface de la peau et donc plus difficilement contrôlables. L'impact de ces perturbations est soit durable, telles que dues à des cicatrices ou de l'usure de la peau, soit temporaire, telles que dues à la température, l'humidité ou de la saleté. L'enjeu principal réside donc dans l'amélioration de l'image afin de reconstituer les parties altérées de l'empreinte. De plus, selon la finesse des caractéristiques étudiées (niveau des caractéristiques), l'impact de ces perturbations peut être critique.

### 5.1.2/ Réseau vasculaire

La mesure du réseau vasculaire dépend de nombreux facteurs optiques. En effet, à la différence du visage ou des empreintes digitales, l'information pertinente est sous cutanée et nécessite un procédé particulier afin de pouvoir observer les vaisseaux sanguins. La figure 5.5 présente le taux d'absorption des trois principaux "chromophores" présents dans le tissu cutané (l'oxyhémoglobine, la mélanine, et l'eau) en fonction de la longueur d'onde de la lumière. L'oxyhémoglobine est la combinaison de l'hémoglobine, protéine responsable du transport de l'oxygène dans le sang, et de l'oxygène (c'est sous cette forme que l'oxygène est amené vers les tissus). La mélanine regroupe les pigments de couleur foncée (du noir jusqu'au brun tirant sur le rouge) se trouvant dans la peau, les poils, les cheveux et la membrane de l'œil, et dont le rôle est de protéger la peau contre les rayonnements ultraviolets.

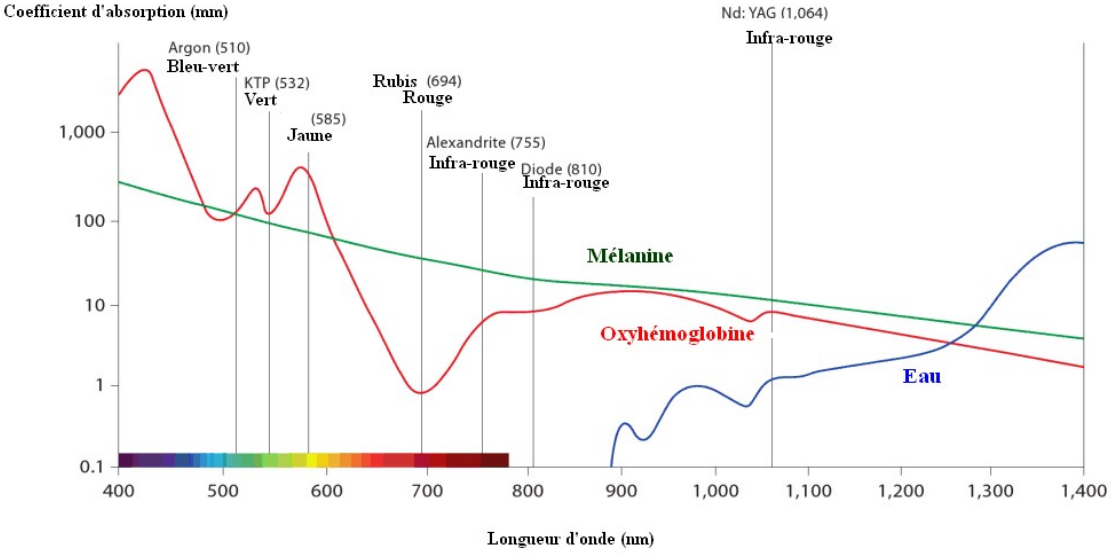



Figure 5.5 – Courbe d'absorption de la lumière par le sang et l'eau [237]

Cette courbe permet d'estimer les longueurs d'ondes optimales permettant l'observation réseau vasculaire en limitant les perturbations visuelles dues à la peau (mélanine) et à l'eau, présente dans tout le corps. D'après cette courbe, on remarque que le sang oxygéné absorbe plus la lumière que la mélanine et l'eau dans les faibles longueurs d'ondes du visible (violet). Autour de 700 nm (lumière rouge), l'oxyhémoglobine n'absorbe plus la lumière contrairement à la mélanine, dont la diminution de l'absorption en fonction de la longueur d'onde est quasi linéaire. L'eau commence à absorber la lumière seulement à 900 nm (proche infrarouge) et dépasse le coefficient d'absorption du sang et de la mélanine autour de 1300 nm. L'absorption de la mélanine et de l'oxyhémoglobine se confondent autour de 900 nm, mais l'absorption de l'oxyhémoglobine reste inférieure à celle de la mélanine.
Cependant, les informations fournies par cette courbe ne suffisent pas. En effet, elle présente ces trois éléments comme étant "côte à côte", sur une surface en champ libre. Or, pour la mesure du réseau vasculaire, ce dernier se trouvant dans le derme, qui se trouve sous l'épiderme, une information de profondeur est à prendre en compte. Comme le montre la figure 5.6, toutes les longueurs d'ondes ne disposent des mêmes facultés pénétrantes dans la peau, et il existe de nombreuses interactions entre les différentes couches organiques et la lumière pouvant perturber la mesure.

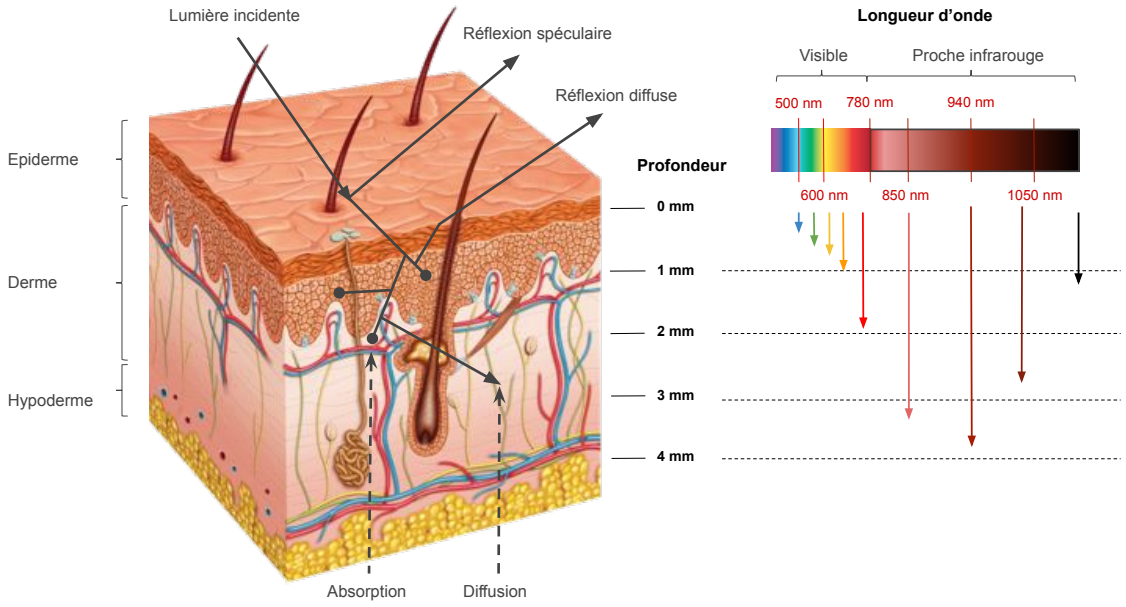



FIGURE 5.6 – Pénétration de la lumière dans la peau selon le spectre [238]

La peau est constituée des trois couches, respectivement l'épiderme, le derme et l'hypoderme. Des vaisseaux superficiels ("capillarités") se trouvent proches de la surface, aux alentours d'1 mm, entre l'épiderme et le derme. Le réseau vasculaire que l'on souhaite observer se trouve plus en profondeur, autour de 4 mm, entre le derme et l'épiderme. De nombreuses interactions se produisent entre la lumière et la peau, jusqu'à atteindre le réseau vasculaire. La lumière incidente est ainsi réfléchie par la surface de la peau, appelée réflexion spéculaire, et une partie pénètre l'épiderme. Après pénétration dans les tissus, la lumière est diffusée dans de multiples directions en profondeur et chaque élément présent sur sa course l'absorbe plus ou moins légèrement. Après diffusion, la lumière remontant à la surface est appelée réflexion diffuse.
Toutes les longueurs d'ondes ne sont pas capables d'atteindre la même profondeur de pénétration. Le spectre visible reste concentré à la surface, bien que les longueurs d'ondes proches du rouge soient légèrement plus pénétrantes, jusqu'aux alentours d'1 mm à 2 mm. Le proche infrarouge (à partir de 780 nm), permet de dépasser les 2 mm de profondeur. Cependant, augmenter la longueur d'onde ne permet pas d'atteindre des couches plus profondes. En effet, le pouvoir pénétrant des longueurs d'ondes au delà de 1100 nm diminue grandement. Il faut donc sélectionner une longueur d'onde répondant à un compromis entre la distance de pénétration dans les tissus et l'absorption des trois éléments de la figure 5.5 (maximiser l'absorption de l'oxyhémoglobine et minimiser l'absorption des autres).

La mesure du réseau vasculaire comme modalité biométrique peut s'effectuer en différents endroits du corps tels que la face intérieure du bras, dos de la main, face intérieure du doigt ou dos du doigt. Le choix de la zone dépend de l'épaisseur de la peau, du type d'acquisition et de la taille du système d'acquisition. Avec une mesure au contact, le doigt sera privilégié avec une acquisition de la face intérieure du doigt, le plus souvent sur la deuxième phalange. Le schéma 5.6 représente les informations génériques de la peau. Au niveau du doigt, la profondeur des trois couches peut être plus fine. Par conséquent, il n'est pas nécessaire d'utiliser une longueur d'onde de 1000 nm pour atteindre le réseau vasculaire.

Notre étude s'oriente sur la mesure du réseau vasculaire pour le doigt. Comme l'illustre la figure 5.7, après une étape d'amélioration d'image pour obtenir le réseau vasculaire, les caractéristiques extraites sont assez proches des caractéristiques de niveau 2 des empreintes digitales. Ces points caractéristiques sont les bifurcations et croisements des lignes formées par le motif du réseau vasculaire. La position des fins de lignes ainsi que les lignes elles mêmes peuvent aussi être extraites.

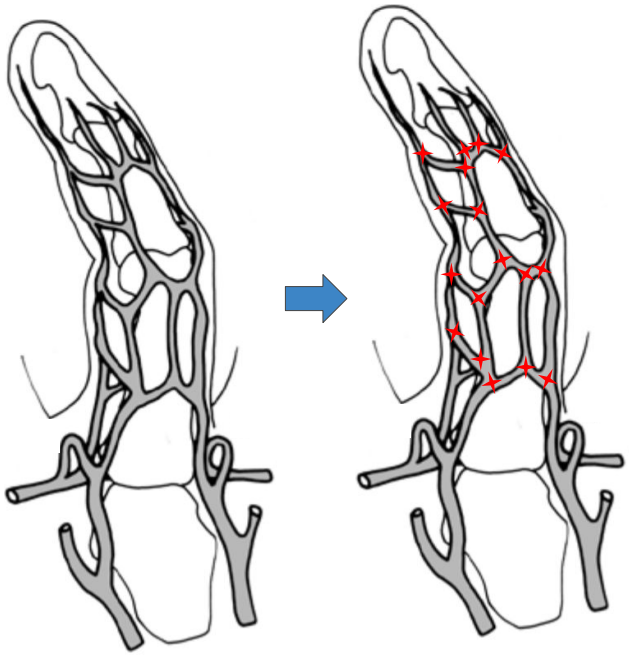


FIGURE 5.7 – Schématisation des points caractéristiques du réseau vasculaire [1]

Les vaisseaux peuvent être mesurés soit sur la face dorsale soit sur la face palmaire du doigt. Dans leur étude [239], Kauba et.al. mettent en lumière les principales différences entre les mesures de ces deux cotés. Il en ressort que les meilleures performances générales sont obtenues pour les images issues de la face palmaire. Cependant, les images dorsales disposent d'informations plus riches, principalement du fait que ces images contiennent le motif du réseau vasculaire, mais également les informations de texture de la peau.

Comme les empreintes digitales, la mesure du réseau vasculaire peut être perturbée par différents facteurs temporaires ou permanents à la surface de la peau (cicatrices, saleté, etc.). D'autres perturbations liées à la température peuvent potentiellement dilater les vaisseaux, ou perturber les propriétés optiques des tissus et donc les interactions entre la lumière et ces derniers. Ces perturbations peuvent par conséquent altérer le motif du réseau vasculaire.
Les principales difficultés résident donc dans l'amélioration (correction) de l'image afin de faire apparaître au mieux le réseau vasculaire. Les nombreuses interactions de la lumière en milieu sous cutané ne permettent pas d'obtenir un contraste suffisant. Cependant, ce contraste peut être amélioré dès l'acquisition en sélectionnant la longueur optimale, comme l'ont montré à la fois Zharov et. al. [240], et Waluś et.al. [241], établissant que le meilleur contraste pour la face intérieure de l'avant bras est obtenu autour de 950 nm, mais que le meilleur compromis se trouve entre 880 nm et 900 nm au vu du faible rendement quantique de la caméra visible CCD utilisée lors de leurs expérimentations.

1. Image source : www.anato.info - "Manuel de chirurgie plastique, reconstructrice et esthétique"

## 5.2/ Méthodes et protocoles

### 5.2.1/ Base de données

Pour réaliser cette étude, nous utilisons des images issues de bases de données mises à disposition par les chercheurs de ce domaine. Les images de réseaux vasculaires du doigt sont issues de la base de données biométrique SDUMLA-HMT (*Shandong University Machine Learning and Applications*) [242]. Cette base est formée d'images de diverses modalités biométriques provenant de 106 individus distincts. La partie correspondant aux réseaux vasculaires contient, par personne, 6 images acquises sans contact à une longueur d'onde de 890 nm par un capteur CCD. Le système d'acquisition contraint le doigt du sujet en translation, mais inclut de légères rotations pouvant être dans le plan ou suivant l'axe du doigt. La figure 5.8 présente des échantillons de cette base. Ces images sont de dimension $320 \times 240$.

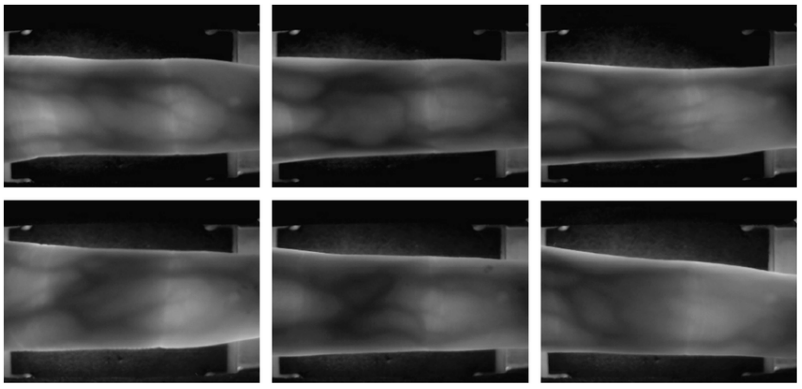


Figure 5.8 – Échantillons de la base de données du réseau vasculaire "SDUMLA-HMT" [242]

Les images d'empreintes digitales que nous utilisons dans cette étude proviennent de la base de données "The Hong Kong Polytechnic University Contactless 2D to contactBased 2D fingerprint" [243]. Cette base contient des images, de dimension $350 \times 225$ acquises d'une part avec contact et sans contact d'autre part. Pour chacun des 300 sujets, la base contient 12 images par type d'acquisition. Afin de nous rapprocher du système que nous souhaitons mettre en place, nous utilisons les images acquises sans contact. Lors de ces acquisitions, la position des doigts des sujets n'ayant été soumise à aucune contrainte, des variations en translation et en rotation sont présentes.

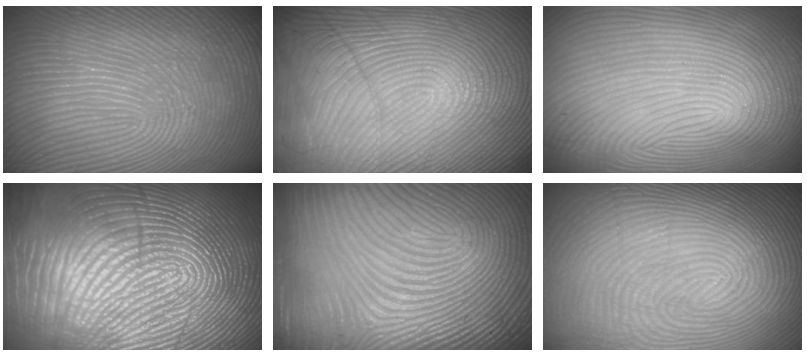
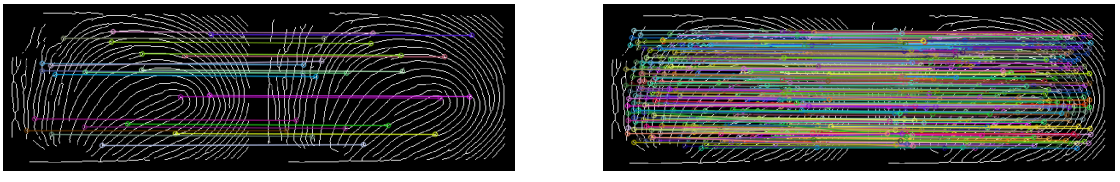


Figure 5.9 – Échantillons de la base de données des empreintes digitales "The Hong Kong Polytechnic University Contactless 2D to contactBased 2D fingerprint" [243]

Par soucis de rapidité de développement et d'évaluation, le nombre d'individus distincts dans chacune de ces bases de données a été restreint. Nous avons isolé les images de 30 personnes pour chacune des deux dernières modalités étudiées. Les images des 15 premières personnes forment notre base d'apprentissage et les images des 15 dernières personnes forment notre base de test.

### 5.2.2/ Principe de l'authentification par mise en correspondance

Pour authentifier une personne, une étude de mise en correspondance entre deux images d'empreintes digitales ou de vaisseaux sanguins est nécessaire. Dans cette étude, l'authentification repose tout d'abord sur le principe classique d'extraction de points d'intérêts et de leur description locale. Puis, les descripteurs obtenus sont mis en correspondances, permettant d'extraire des attributs qui sont ensuite utilisés pour effectuer une décision d'authentification globale. La mise en correspondance des descripteurs locaux est illustrée par les figures 5.10 et 5.11. On observe que, même pour des images de personnes différentes, des correspondances entre des points singuliers ou des points sur des lignes sont inévitables. Cependant, contrairement à deux images d'une même personne, un certain désordre spatial est à noter entre les points considérés comme similaires (grande variance dans les directions des correspondances entre deux images).



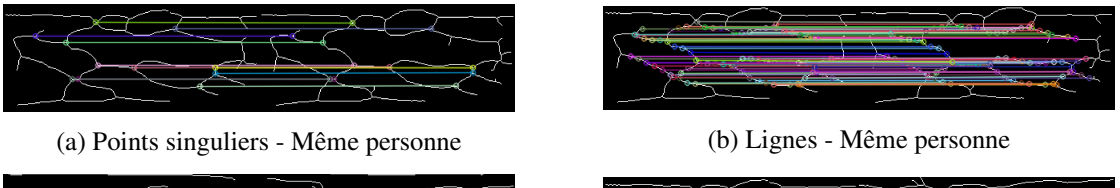
(a) Points singuliers - Même personne

(b) Lignes - Même personne

(c) Points singuliers -Personnes différentes

(d) Lignes - Personnes différentes

Figure 5.10 – Illustration de la mise en correspondance de deux empreintes digitales selon les points singuliers (bifurcations et jonctions) et des points formant les lignes



(a) Points singuliers - Même personne

(b) Lignes - Même personne

(c) Points singuliers - Personnes différentes

(d) Lignes - Personnes différentes

Figure 5.11 – Illustration de la mise en correspondance de deux réseaux vasculaires selon les points singuliers (bifurcations et jonctions) et des points formant les lignes

La mise en correspondance consiste à comparer des caractéristiques de références avec des caractéristiques extraites d'une nouvelle image. Pour obtenir ces caractéristiques, un certain nombre de prétraitements sont nécessaires. Comme le montre les deux figures précédentes, les images d'empreintes digitales et de vaisseaux sanguins utilisées ont une apparence particulière. Cette forme, appelée squelette, est une représentation binaire des structures géométriques de ces modalités. L'image ne contient que deux valeurs de pixels, à savoir 1 (objet) ou 0 (arrière plan). Les structures composant ces motifs sont alors des lignes d'un seul pixel de large. Cette représentation permet de détecter facilement les points d'intérêts utilisés pour la mise en correspondance. Ces points d'intérêts représentent des particularités de ces structures, telles que des bifurcations de lignes, des jonctions de lignes, des fins de lignes ou encore les lignes elles-mêmes.
Ces images squelettisées sont obtenues après plusieurs prétraitements pouvant être divisés en deux étapes principales, comme le montre la figure 5.12. Une première phase d'amélioration d'image permet de réduire le bruit dans l'image et d'améliorer l'image dans le sens où les performances de l'étape suivante sont maximisées. Cette seconde étape consiste en une segmentation où, de par un seuillage, l'image est binarisée en attribuant une certaine valeur aux structures pertinentes et une autre valeur à l'arrière plan. La squelettisation est ensuite appliquée sur le résultat de ce seuillage. La détection de points d'intérêt est suivie d'une étape de description locale de la région de chacun de ces derniers. Pour une image de référence, l'ensemble des vecteurs de description ainsi obtenus représentent l'information "discriminante" utilisée pour comparer deux images.

La phase d'enrôlement consiste donc à acquérir une image de chaque modalité et d'en extraire, pour chacune, certaines caractéristiques (descripteurs) qui seront stockées sur la carte sans contact. Tous les traitements exposés, que nous allons détailler par la suite, sont communs aux empreintes digitales et aux réseaux vasculaires, mais leur paramétrage dépend de la modalité considérée.

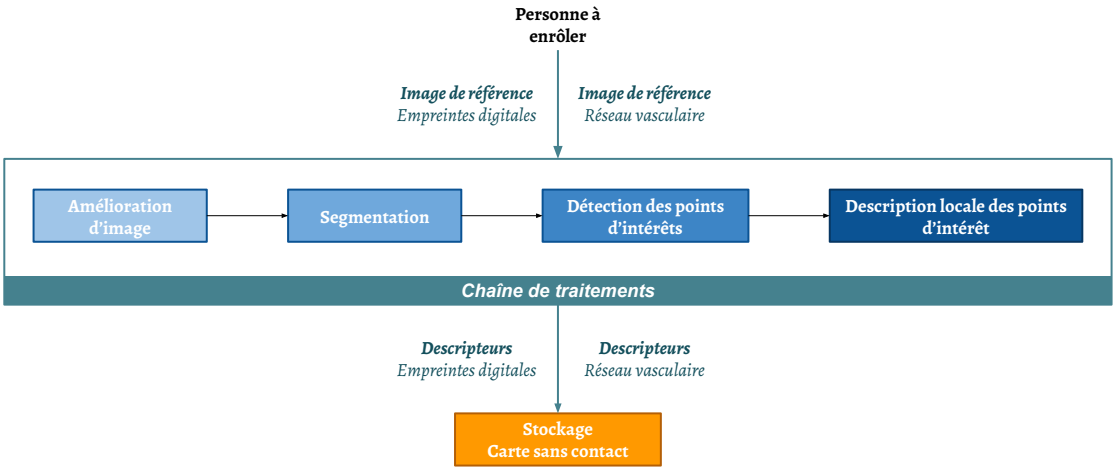



FIGURE 5.12 – Description du protocole de traitement de l'image, communs aux modalités du doigt, pour la phase d'enrôlement

La phase d'authentification est décrite par la figure 5.13. Une personne se présente devant le système et une image de chaque modalité étudiée est acquise. Les traitements précédents lui sont appliqués afin d'en extraire les descripteurs correspondant à chaque image. Ces derniers sont alors mis en correspondance avec les descripteurs de référence, stockés sur la carte sans contact. La décision d'authentification est ensuite effectuée par un classifieur. Pour l'authentification de visages, l'espace de visage (produit par l'ACP), permettant de décrire l'image, est commun à chaque individu et est stocké localement. L'authentification est alors assurée par un classifieur "général" stocké sur la plateforme de calcul.

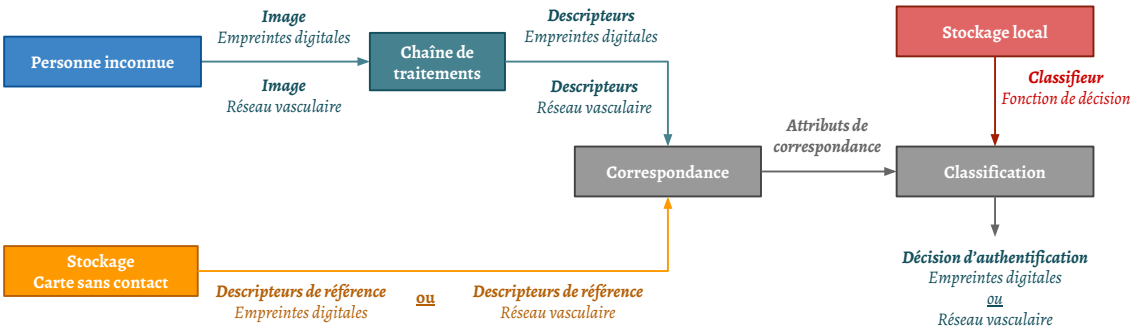



FIGURE 5.13 – Description du protocole d'authentification par mise en correspondance des empreintes digitales et des vaisseaux

Ici, la description d'une image est individuelle et est stockée sur la carte sans contact. L'authentification est assurée par un classifieur, commun à chaque individu et stocké localement. Ce classifieur, dont la mise en place est illustrée par la figure 5.14, a pour objectif d'établir une décision d'authentification en fonction de divers attributs issus de la correspondance entre les descripteurs de deux images. La mise en correspondance fournit un grand nombre de données, et une sélection des attributs optimaux (fournissant la plus faible erreur) est effectuée par un algorithme de sélection d'attributs SBFS (*Sequential Backward Floating Selection*).

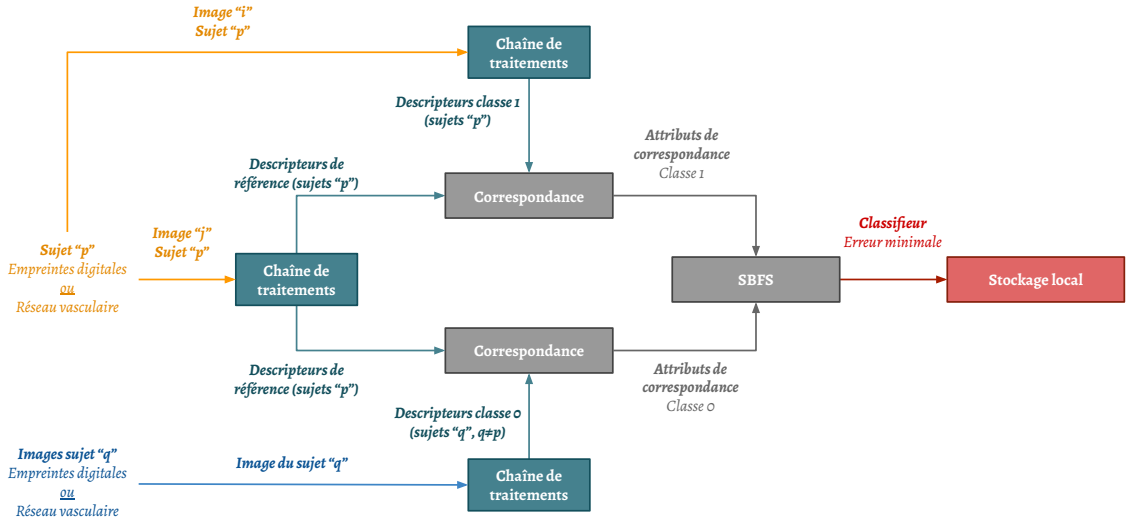



FIGURE 5.14 – Description de la création du classifieur pour la décision globale en fonction des attributs de mise en correspondance

La SBFS permet de sélectionner, tel que le montre la figure 5.15, un sous ensemble optimal $U_k$ de $d'$ attributs, parmi un ensemble $Y$ de $d$ attributs, répondant à un certain critère défini par l'utilisateur. Ce critère est une fonction de coût, $J$, obtenue par l'évaluation des performances sur ces attributs (classification, mesure de distance, etc). Pour une itération, la SBFS évalue toutes les fonctions de coûts relatives à toutes les combinaisons possibles dans le cas où un et un seul attribut $f_i$ est retiré. Ainsi, pour tout attribut $f_i$ parmi les $d'$ attributs de l'itération, la SBFS produit la fonction de coût $J(U_k - f_i)$. Dans le cas où celle-ci est plus avantageuse que la fonction de coût de référence, la SBFS conserve le sous-ensemble optimal où cet attribut a été retiré. L'itération

suivante consiste à effectuer le même protocole sur le nouveau sous ensemble où un attribut a été retiré. La SBFS que nous avons implanté itère ce processus jusqu'à obtenir un ensemble d'attributs nul. À la fin de la recherche, la SBFS fourni alors le sous ensemble $U_{fin}$ ayant fourni la meilleure fonction de coût $J_{min} = J(U_{fin})$ parmi toutes les combinaisons d'attributs possibles.

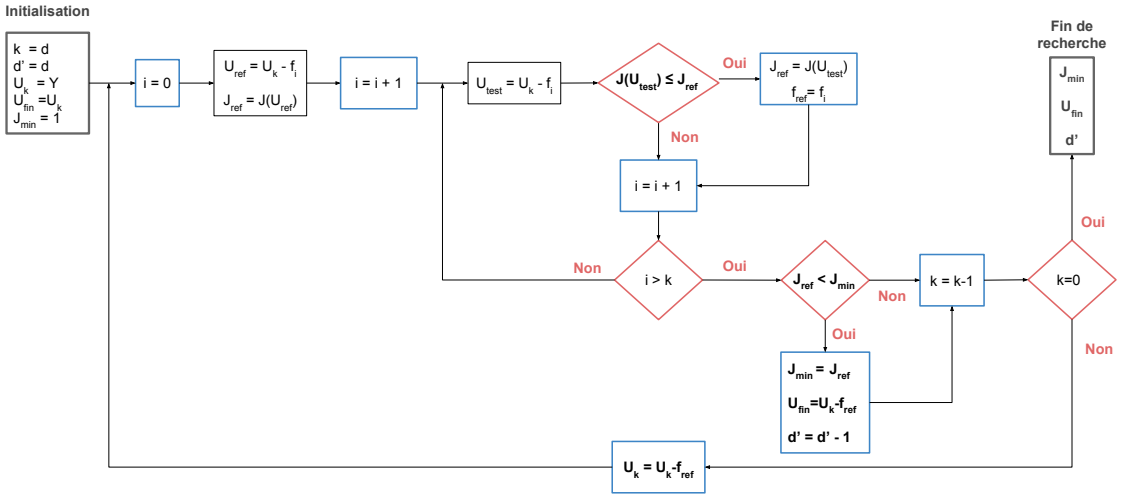



FIGURE 5.15 – Schéma fonctionnel de la recherche d'attributs sous-optimaux par SBFS

L'étude présentée dans ce chapitre traite des performances brutes de chaque modalité considérée indépendamment. La section 6.1.2 du chapitre suivant traite de l'authentification basée sur une fusion de décisions à partir du traitement du visage, des empreintes et des vaisseaux. Dans la suite de cette section, nous présentons les divers algorithmes utilisés dans chaque phase de prétraitement, ainsi que les techniques de description et de classification utilisées.

#### 5.2.2.1/ AMÉLIORATIONS DE L'IMAGE

Une approche populaire dans la littérature, permettant l'amélioration des images d'empreintes digitales ainsi que des vaisseaux, repose sur l'utilisation de filtres de Gabor. Largement utilisé en analyse de formes, ce filtre permet de favoriser efficacement les caractéristiques des objets dans l'image. Il s'agit de filtres passe-bandes ayant des propriétés sélectives en orientation et en fréquence, constitués d'une onde sinusoïdale plane modulée par une enveloppe gaussienne. En d'autres termes, ils peuvent être paramétrés de manière optimale à une fréquence et une orientation spécifique, permettant d'obtenir une réponse maximale au niveau des structures linéaires tout en réduisant le bruit dans l'image.
Cependant, ce filtre nécessite un paramétrage précis. Pour les empreintes digitales, différents paramètres sont déterminés localement afin d'extraire le plus efficacement possible toute la variabilité des crêtes. L'estimation de ces différents paramètres optimaux requiert par conséquent une image de bonne qualité, ce qui n'est pas toujours le cas. Pour les empreintes, les images peuvent contenir du bruit d'acquisition, de même que diverses perturbations dues à l'éclairage par exemple. Nous avons vu également que les images de vaisseaux étaient très faiblement contrastées du fait des nombreuses dispersions lumineuses dans les tissus, ainsi que de l'absorption à différents niveaux. Il est donc nécessaire de considérer une étape préliminaire de "correction" d'image.

### Étape préliminaire : extraction de la région d'intérêt des vaisseaux

D'après les échantillons des bases de données, il est notable que, contrairement aux empreintes digitales, les images du réseau vasculaire ne sont pas formées uniquement de cette modalité. En effet, elles présentent également une partie du système d'acquisition en arrière plan. Par conséquent, pour limiter la prise en compte de ce dernier dans la segmentation, une étape préliminaire d'extraction de zone d'intérêt du doigt est nécessaire. Nous proposons une méthode permettant cette extraction, cependant, celle-ci est exclusivement basée sur les images de la base de données que nous utilisons, car elle dépend des informations propres au système d'acquisition. On observe que le doigt est centré sur l'image, mais sa position varie entre les différents clichés. L'extraction d'une zone fixe n'est donc pas une option pertinente.

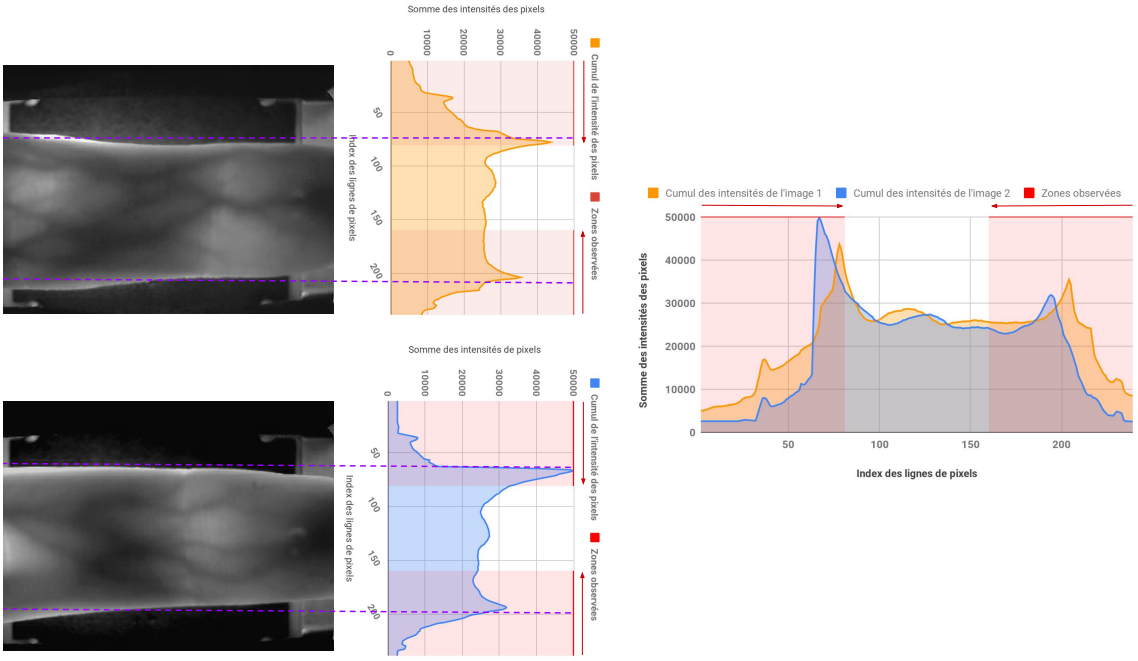
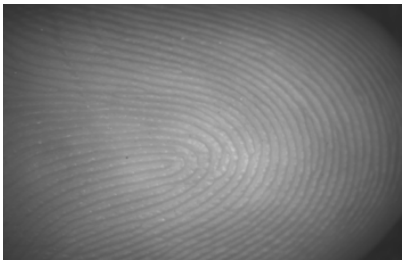


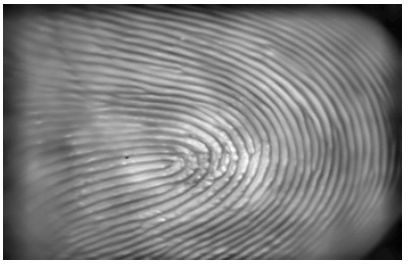

FIGURE 5.16 – Extraction de la zone d'intérêt du doigt pour la base "SDUMLA-HMT"

De par l'étude des différentes images, nous avons choisi d'observer les pixels contenus dans deux zones de l'image : le premier et le dernier tiers, correspondant donc aux 80 premières et dernières lignes de l'image. Notre méthode se base sur la somme des intensités lumineuse des pixels pour chaque ligne. L'intensité lumineuse des contours du doigt étant plus grande que l'arrière plan, deux pics sont notables sur le graphe, figure 5.16, représentant cette somme pour chaque ligne de l'image. La zone d'intérêt à extraire est donc délimitée par la partie de l'image comprise entre ces deux pics. Nous définissons deux limites, chacune basée sur la plus grande différence entre la somme de deux lignes consécutives, en partant de chaque extrémité de l'image. En d'autres termes, ces limites correspondent aux plus grands fronts montants et descendants pour respectivement la zone supérieure et la zone inférieure de l'image. Afin de conserver une certaine souplesse dans cette détection de région d'intérêt, nous sélectionnons la cinquième ligne avant la première ligne détectée, et la cinquième ligne après la deuxième détectée. La superposition des graphes représentant la somme des intensités de chaque ligne, pour deux images différentes de la base de donnée, nous montre que ces deux pics sont bien contenus dans les deux zones observées par notre méthode.

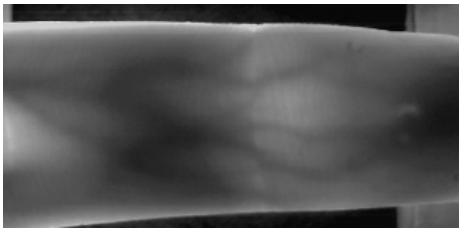
**Étage 1 - Correction d'image :**
Selon Olsen et. al. [244], la méthode "CLAHE" *(Contrast Limited Adaptive Histogram Equalization)* est particulièrement adaptée à l'amélioration des images de réseaux vasculaires. Présentée en annexe B.2.1.2, cet algorithme permet d'effectuer une égalisation adaptative d'histogramme avec l'introduction d'un paramètre permettant de limiter l'augmentation de contraste produite localement. En d'autres termes, le nombre de pixels associés à chaque valeur d'intensité lumineuse dans l'histogramme local est tronqué à la valeur de ce paramètre. Après cette troncature d'histogramme, les pixels dépassant ce seuil sont redistribués également au travers de l'histogramme. Principalement utilisé sur des images du domaine médical (de type IRM), cette méthode représente un bon compromis entre la complexité de calculs et la qualité de l'amélioration d'image. En revanche, sur des images bruitées, l'application de CLAHE conduit à augmenter la proportion de ce bruit. Nous proposons alors d'introduire une phase de débruitage, précédant CLAHE.
Les méthodes traditionnelles de débruitage par lissage sont appliquées par fenêtrage et ont pour objectif de réduire le bruit dans l'image tout en conservant les principaux aspects géométriques. Cependant, elles ne permettent pas de préserver les structures fines, les détails et les textures. Par ailleurs, les empreintes digitales présentent une structure formée de lignes fines et relativement proches. Ces lignes étant l'objet de notre attention, il est nécessaire d'appliquer un filtrage conservant au maximum les hautes fréquences propres aux contours de ce motif. Il en va de même pour le réseau vasculaire. Bien que ce dernier soit formé de structures linéaire éloignées, celles-ci sont "estompées". Il s'agit donc de ne pas les atténuer tout en cherchant à réduire le bruit. Nous proposons alors l'application d'une méthode de débruitage par moyennes non locales [245], laquelle nous semble particulièrement adaptée aux deux modalités traitées. Cet algorithme exploite avantageusement le haut degré de redondance des différentes structures dans une image. En effet, selon Buades et.al. [246], chaque fenêtre centrée en un pixel dans l'image possède au moins une fenêtre similaire dans cette image. Cette méthode, pour laquelle plus de précisions peuvent être trouvées en annexe B.2.1.1, est donc capable de réduire le bruit dans l'image tout en conservant un maximum de composantes dans les hautes fréquences.



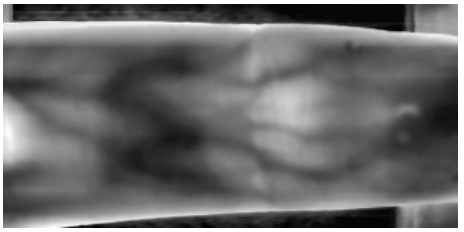
(a) Empreintes - Région d'intérêt - $I(x, y)$



(b) Empreintes - Correction d'image - $C(x, y)$



(c) Vaisseaux - Région d'intérêt - $I(x, y)$



(d) Vaisseaux - Correction d'image - $C(x, y)$

FIGURE 5.17 – Images d'empreintes digitales et de réseau vasculaire résultantes de la correction d'images (Débruitage par moyennes non locales et amélioration de contraste par CLAHE)

**Étage 2 - Filtre de Gabor :**

Nous présentons ici le filtre de Gabor et son application concrète aux empreintes digitales et aux réseaux vasculaires. La partie réelle du filtre, définie par l'équation 5.1 et illustrée par la figure 5.18, est construite par une onde cosinus modulée par une gaussienne. Elle est souvent utilisée pour extraire des caractéristiques pertinentes dans des images à des fins biométriques. Une fois les paramètres du filtre $G$ définis, l'image $F(x, y)$ résulte de la convolution de ce filtre en tout point $(x, y)$ d'une image $I$, tel que $F(x, y) = G(x, y) * I(x, y)$.

$$G(x, y; \lambda, \theta, \psi, \sigma, \gamma) = exp\left(-\frac{x_\theta^2 + \gamma^2.y_\theta^2}{2.\sigma^2}\right).cos\left(2.\pi.\frac{x_\theta}{\lambda} + \psi\right) \tag{5.1}$$

où :

- $x_\theta = x.cos(\theta) + y.sin(\theta)$
- $y_\theta = -x.sin(\theta) + y.cos(\theta)$
- $\lambda$ représente la longueur d'onde du filtre ($\lambda = 1/f$, où $f$ est la fréquence)
- $\theta$ représente l'orientation de la normale de l'onde du noyau
- $\psi$ est un décalage de phase
- $\sigma$ est la déviation standard de l'enveloppe de la gaussienne
- $\gamma$ est rapport d'aspect spatial

Les motifs décrits par les empreintes digitales et le réseau vasculaire sont tous deux formés de lignes courbes. Cependant, leur aspect général et leur répartition dans l'images sont différents. En conséquence, nous n'appliquons pas cette étape de filtrage de la même manière à ces images. Les empreintes digitales présentent des lignes dont les orientations varient rapidement au travers de l'image, et dont la fréquence peut également varier localement. C'est pourquoi, l'image des empreintes est divisée en régions où sont estimées l'orientation et la fréquence des lignes [173]. Pour chaque région, un filtre de Gabor est paramétré d'après les estimations de l'orientation et de la fréquence locales. Pour le réseau vasculaire, la fréquence des lignes est basse mais les orientations varient fortement. Une banque de filtres de Gabor est utilisée. Cette banque comporte différents noyaux dont seule l'orientation varie. Afin d'éviter de calculer les orientations localement, ils sont tous appliqués en tout point de l'image et les pixels sont sélectionnés parmi les images résultantes suivant une certaine règle appliquée à leurs valeurs.

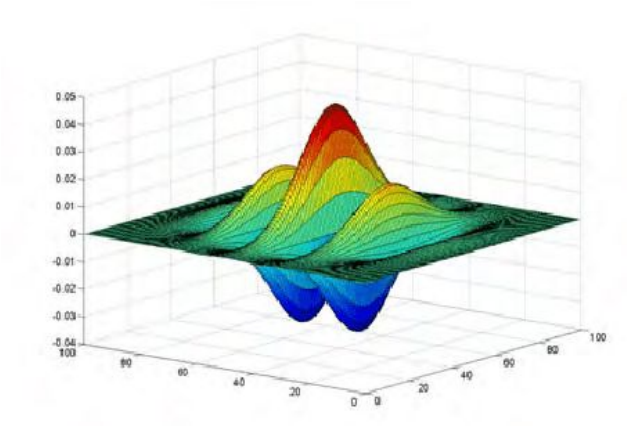


FIGURE 5.18 – Aperçu d'un noyau de filtre de Gabor pour une certaine configuration

**Empreintes digitales :**
Dans le cadre de l'amélioration des images d'empreintes digitales, l'estimation des orientations locales des lignes est une étape cruciale, permettant de paramétrer au mieux le filtre de gabor afin d'obtenir une réponse locale optimale. La carte des orientations d'une empreinte digitale définit l'orientation locale des lignes de l'empreinte par rapport à l'axe horizontal. Les étapes de cette estimation sont les suivantes [247] :

Soit un bloc de taille $W_\theta \times W_\theta$, centré en un pixel $(i, j)$ de l'image améliorée d'empreintes digitales. Pour chaque pixel du bloc, les gradients de direction $x$ et $y$ sont calculés. Leurs magnitudes, nommées respectivement $\partial_x(i, j)$ et $\partial_y(i, j)$, sont obtenues par l'application des opérateurs de Sobel horizontaux et verticaux, décrits dans la section 4.2.1.1. L'orientation locale au pixel $(i, j)$, nommée $\theta(i, j)$, peut alors être estimée de par l'utilisation du système d'équations suivant :

$$\left.\begin{aligned} V_x(i,j) &= \sum_{u=i-\frac{W_\theta}{2}}^{i+\frac{W_\theta}{2}} \sum_{v=j-\frac{W_\theta}{2}}^{j+\frac{W_\theta}{2}} 2.\partial_x(u,v).\partial_y(u,v) \\ V_y(i,j) &= \sum_{u=i-\frac{W_\theta}{2}}^{i+\frac{W_\theta}{2}} \sum_{v=j-\frac{W_\theta}{2}}^{j+\frac{W_\theta}{2}} \partial_x^2(u,v).\partial_y^2(u,v) \end{aligned}\right\} \Rightarrow \theta(i,j) = \frac{1}{2}\tan^{-1}\frac{V_x(i,j)}{V_y(i,j)} \tag{5.2}$$

L'étape suivante consiste à lisser ces orientations locales, à l'aide d'un filtre Gaussien, afin de corriger les zones où les orientations varient brutalement ou sont discontinues. La carte des orientations est alors convertie en un champ vectoriel où, suivant l'équation 5.3, $\phi_x$ et $\phi_y$ sont respectivement les composantes $x$ et $y$. Le système d'équations 5.4 définit ensuite l'application du filtre, où $G$ est le filtre Gaussien passe bas de taille $W_\phi \times W_\phi$. La carte des orientations ainsi lissées est définie par $O(i, j)$.

$$\begin{aligned} \phi_x &= \cos(2\theta(i,j)) \\ \phi_y &= \sin(2\theta(i,j)) \end{aligned} \tag{5.3}$$

$$\left.\begin{aligned} \phi'_x(i,j) &= \sum_{u=-\frac{W_\phi}{2}}^{\frac{W_\phi}{2}} \sum_{v=-\frac{W_\phi}{2}}^{\frac{W_\phi}{2}} G(u,v)\phi_x(i-u,j-v) \\ \phi'_y(i,j) &= \sum_{u=-\frac{W_\phi}{2}}^{\frac{W_\phi}{2}} \sum_{v=-\frac{W_\phi}{2}}^{\frac{W_\phi}{2}} G(u,v)\phi_y(i-u,j-v) \end{aligned}\right\} \Rightarrow O(i,j) = \frac{1}{2}.\tan^{-1}\frac{\phi'_x(i,j)}{\phi'_y(i,j)} \tag{5.4}$$

La seconde étape du paramétrage local du filtre de Gabor consiste à calculer la fréquence locale des lignes de l'empreinte. De la même manière que pour l'estimation des orientations locales, l'image est divisée en de nouveaux blocs de taille $W_F \times W_F$. S'en suit une projection des intensités des pixels suivant la direction orthogonale à l'orientation locale. Le résultat de cette projection est une onde sinusoïdale où les minimums locaux représentent les lignes de l'empreinte. La longueur d'onde de cette sinusoïde, notée ici $S(i, j)$, est représentée par l'écart entre les lignes qui est calculée est comptant le nombre médian de pixels situés entre deux minimums consécutifs. La fréquence d'un bloc centré en un pixel $(i, j)$, notée $F(i, j)$, est définie par l'équation 5.5.

$$F(i,j) = \frac{1}{S(i,j)} \tag{5.5}$$

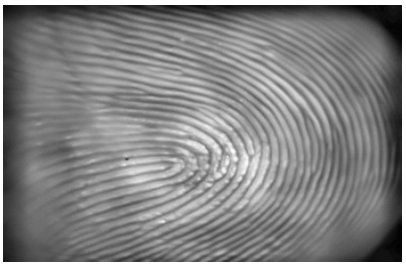
Dans un bloc, lorsque la projection n'est pas explicitement exploitable, ou lorsque qu'une minutie est présente, l'estimation de la fréquence n'est pas considérée comme fiable. Dans ce cas, la fréquence est interpolée à partir des fréquences des blocs voisins. Le filtre de Gabor $G$ est ensuite appliqué à l'image améliorée $C$ par convolution et est paramétré localement suivant les estimations des orientations $O(i, j)$ et des fréquences $F(i, j)$ locales, comme le spécifie l'équation 5.6 où $w$ représente la taille du noyau du filtre .

$$E(i,j) = \sum_{u=-\frac{w}{2}}^{\frac{w}{2}} \sum_{v=-\frac{w}{2}}^{\frac{w}{2}} G(u,v\,;\,O(i,j),F(i,j))C(i-u,j-v) \tag{5.6}$$

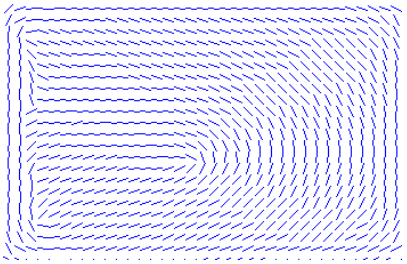
La bande passante du filtre, définissant la plage de réponse en fréquences, est fixée par le paramétrage de sa déviation standard $\sigma$. L'objectif étant de paramétrer le filtre pour correspondre à la fréquence locale les lignes de l'empreinte, ce paramètre doit être fonction de cette fréquence locale. Cette déviation standard est par conséquent pondérée par un facteur constant $k$ tel que défini par l'équation 5.7. De plus, pour permettre au filtre d'ajuster sa taille en fonction de cette bande passante, Raymond Thai propose de définir $w$ la taille du filtre selon cette déviation standard [247], tel qu'illustré par l'équation 5.8.

$$\sigma = k.F(i,j) \tag{5.7}$$

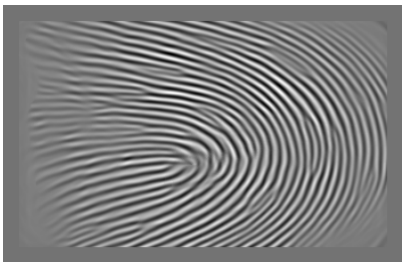
$$w = 6.\sigma \tag{5.8}$$



(a) Image corrigée - $C(x, y)$



(b) Carte des orientations - $O(x, y)$



(c) Filtrage de Gabor - $E(x, y)$

FIGURE 5.19 – Images des empreintes digitales issues du filtrage par la banque de filtres de Gabor

**Réseau vasculaire :**
L'utilisation du filtre de Gabor pour l'amélioration du réseau vasculaire du doigt diffère de l'implémentation pour les empreintes digitales. Le réseau vasculaire observé dans le proche infrarouge étant formé de structures plus grossières, il n'est pas nécessaire de paramétrer aussi finement le filtre. Après avoir fixé les paramètres optimaux du filtre (exposés dans l'équation 5.1) pour une image disposant de certaines caractéristiques, Lu et.al [248] proposent de créer une banque de

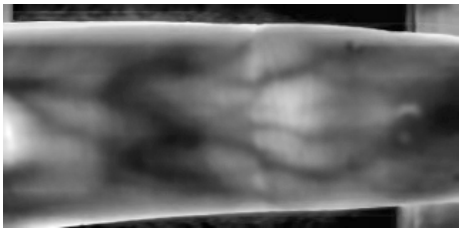
filtres dont seule l'orientation varie. Les autres paramètres étant fixes, $K$ filtres $G_k$ sont créés en faisant varier l'orientation $\theta$ dans la plage $[\pi/K; \pi]$. Chaque noyau est normalisé en lui divisant la somme de ses éléments, et est alors noté $G_k^N$. Ainsi, si l'on considère $F_k$ l'image résultant de la convolution entre le filtre $G_k^N$ (correspondant à l'orientation $k$) et l'image corrigée $C$, $F_k(x, y)$ est obtenu suivant l'équation 5.9.

$$\left.\begin{array}{rl} G_k^N & = \dfrac{G_k}{\sum_{i=0}^{w}\sum_{j=0}^{w} G_k(i,j)} \\ \theta_k & = k\dfrac{\pi}{K} \end{array}\right\} \Rightarrow \quad F_k(x,y) = G_k^N * C(x,y), \qquad \forall k \in \{1;2;...;K\} \tag{5.9}$$

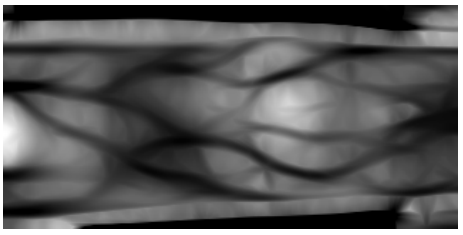
Chaque image $F_k$ résultant de cette convolution contient la réponse du filtre de Gabor selon une certaine orientation $\theta_k$. Les intensités de ces réponses sont minimales lorsque l'orientation locale d'une veine correspond à l'orientation du filtre, et maximales dans le cas contraire. Par conséquent, l'image optimale $F^{'}$ " du réseau vasculaire est obtenue en sélectionnant chaque pixel $(x, y)$ suivant une règle de valeur minimale parmi les K images $F_k$ (équation 5.10). Pour finir, afin de limiter l'apparition de certaines hautes fréquences indésirables, une nouvelle image $E$ est construite à partir d'une somme pondérée de l'image résultante et de l'image originale $I$ (équation 5.11). L'annexe B.2.2.3 illustre les images résultant des orientations de la banque de filtres de Gabor.

$$F^{'}(x,y) = \min[F_k(x,y)], \qquad \forall k \in \{1;2;...;K\} \tag{5.10}$$

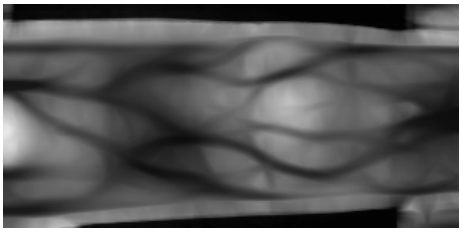
$$E(x,y) = \beta.F^{'}(x,y) + (1-\beta).I(x,y), \quad \beta = 0,7 \tag{5.11}$$



(a) Image corrigée - $C(x, y)$



(b) Image filtrée par le filtre de Gabor - $F'(x, y)$



(c) Somme pondérée de l'image filtrée avec l'image originale - $E(x, y)$

FIGURE 5.20 – Images des vaisseaux issues du filtrage par la banque de filtres de Gabor

**En résumé :**

Les figures 5.21 et 5.22 résument respectivement le protocole de la phase d'amélioration d'images d'empreintes digitales et d'images de réseaux vasculaires. Elle comprend une étape de débruitage assurée par la méthode des moyennes non locales, suivie par une égalisation d'histogramme avec une limitation de contraste (CLAHE). L'image est alors filtrée par de multiples filtres de Gabor, pour lesquels les paramètres sont ajustés en fonction de la modalité.

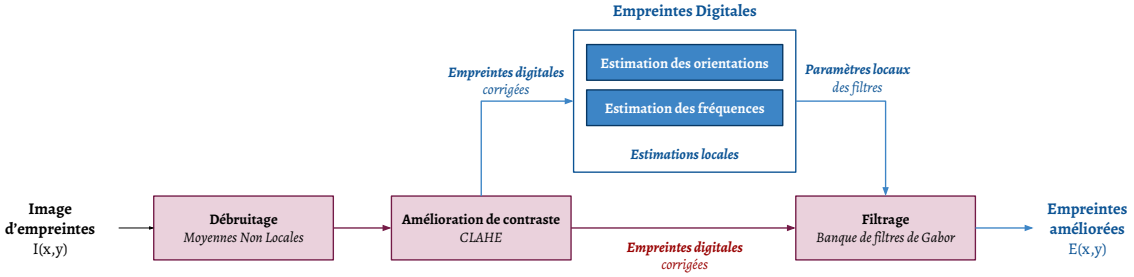



FIGURE 5.21 – Description du protocole global de la phase d'amélioration des d'empreintes

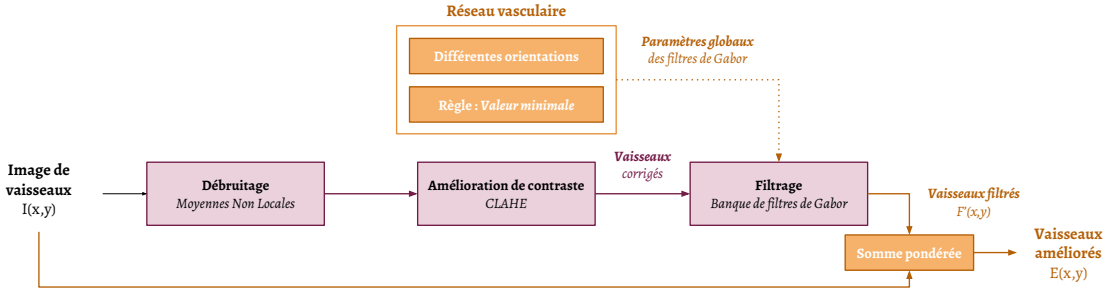



FIGURE 5.22 – Description du protocole global de la phase d'amélioration des images de Vaisseaux

### 5.2.2.2/ SEGMENTATION

Les traitements suivants consistent à segmenter l'image afin de faciliter l'extraction des caractéristiques. Comme le montre l'image 5.23, cette tâche est découpée en différentes étapes successives communes aux deux modalités. Une phase de correction d'images est à nouveau appliquée. Celle-ci est toujours composée d'une étape de débruitage et de correction de contraste. Cependant, de nouveaux paramètres sont appliqués. Cette nouvelle correction d'images permet d'obtenir un seuillage plus stable, pour lequel différentes méthodes sont étudiées.

L'image binaire issue de cette étape de seuillage permet d'effectuer la squelettisation. Toujours dans l'optique de garantir la plus grande stabilité dans les traitements, une phase de nettoyage de l'image binaire est effectuée. Celle-ci permet de "lisser" les contours de la structure obtenue, permettant ainsi de limiter la création de lignes indésirables dans le squelette. La figure 5.24 illustre les résultats cette nouvelle phase de correction d'images pour les deux modalités.

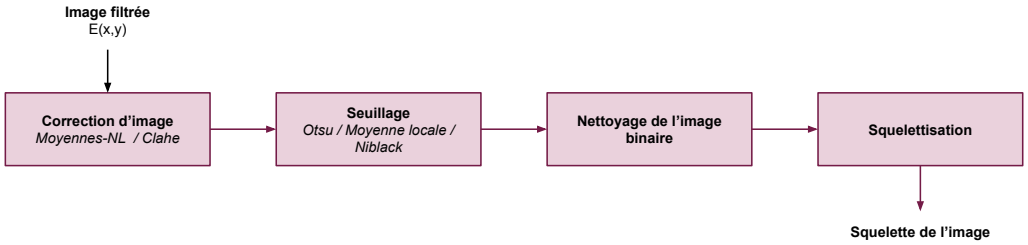
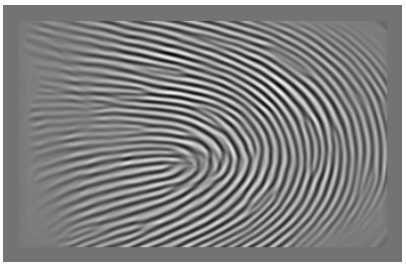
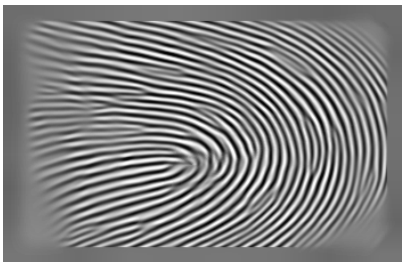


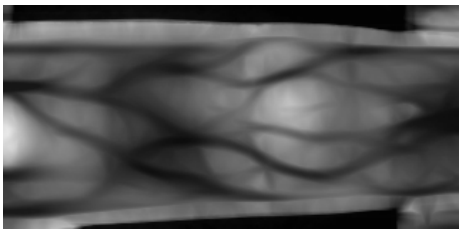

FIGURE 5.23 – Description du protocole de la phase de segmentation



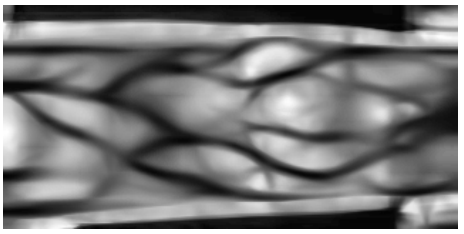
(a) Empreintes filtrée - $E(x, y)$



(b) Empreintes filtrées et corrigées - $E_C(x, y)$



(c) Vaisseaux filtrés - $E(x, y)$



(d) Vaisseaux filtrés et corrigés - $E_C(x, y)$

FIGURE 5.24 – Images d'empreintes digitales et de réseau vasculaire résultantes de la correction d'images $E_C$ appliquée à l'image en sortie du filtre de Gabor $E$

**Seuillage :**
Le seuillage permet de "binariser" une image de telle sorte qu'une valeur de luminance représente "l'objet" et une autre représente "l'arrière plan". Dans le cas général, chaque intensité de pixel est comparée à un seuil, défini pour toute l'image. Les pixels dont les valeurs sont supérieures à ce seuil sont considérés comme faisant partie de l'objet. Le seuil peut être défini manuellement ou déterminé de manière automatique. Une des techniques automatiques les plus utilisées dans la littérature est la méthode d'Otsu [249]. L'objectif de cette méthode est de déterminer un seuil optimal séparant l'image en deux classes (objet et arrière plan), de telle sorte que la variance de chaque classe soit minimisée. Néanmoins, les valeurs des pixels représentant l'information pertinente pouvant être proches des valeurs de l'arrière plan, voir confondues, la distribution des intensités lumineuse n'est pas toujours distincte dans l'image. L'utilisation d'un seuillage adaptatif local permet de résoudre ce problème. Cette méthode considère un seuil non plus global mais local, dépendant de données contenues dans une fenêtre glissante. Cela permet de prendre en compte la grande variabilité des pixels formant l'information.
Il existe de très nombreuses façons d'ajuster ce seuil [250]. Nous nous intéresserons ici à seulement deux seuils, lesquels nous ont semblé fournir un bon compromis entre complexité de calculs et qualité de seuillage. A titre d'information, d'autres techniques de seuillage sont présentés en annexe B.2.1.4. Dans les équations suivantes, la notation $\overline{W}_{(x,y)}$ définit la valeur moyenne des intensités de pixels dans une fenêtre $W$ centrée en $(x, y)$ et le paramètre $c$ représente une constante permettant d'ajuster la valeur du seuil. Dans la suite de ce chapitre, nous comparerons ces deux seuils locaux, ainsi que le seuillage d'Otsu.

— **Seuillage par moyennes locales :**

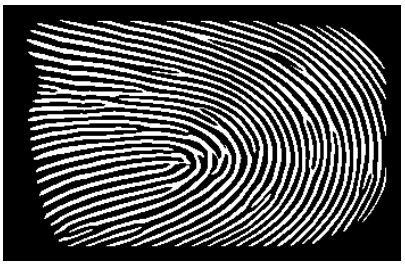
$$Seuil_{W_{(x,y)}} = \overline{W}_{(x,y)} - c \tag{5.12}$$

— **Seuillage de Niblack :**
Le seuillage de Niblack [251] considère en plus la variance $\sigma(W_{(x,y)})$ du voisinage du pixel, laquelle est alors contrôlée par un paramètre $k$ compris dans l'intervalle $[-1; 1]$.

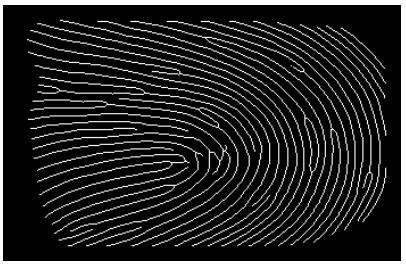
$$Seuil_{W_{(x,y)}} = \overline{W}_{(x,y)} + k.\sigma(W_{(x,y)}) - c \tag{5.13}$$

**Affinage des structures :**
Selon le seuillage utilisé et la complexité de l'image (répartition des intensités de pixels entre l'objet et l'arrière plan), il peut arriver que certains artéfacts non désirés soient considérés comme pertinents. Cela résulte en des zones isolées souvent plus petites que l'objet à segmenter. Par conséquent, nous proposons d'effectuer un "nettoyage" à partir d'une détection de contours. Les contours détectés dont la surface est inférieure à un certain seuil sont alors supprimées. Nous définissons manuellement ce seuil mais il pourra être calculé automatiquement au travers de plus amples expérimentations. Les contours des structures seuillées de l'image ne sont pas formés de lignes régulières du fait des variations brutales des valeurs des pixels. Une étape d'affinage consiste à "lisser" ces contours en supprimant les pixels responsables de cette "rugosité". Cette tâche est réalisée par une fenêtre glissante de taille 3 x 3 se déplaçant avec un pas de 1 pixel. Si ses "coins" sont nuls, la valeur des autres pixels est vérifiée. Si et seulement si un seul d'entre eux est non nul, le pixel correspondant à sa position dans l'image est mis à zéro.

**Squelettisation :**
L'étape finale avant l'extraction de caractéristiques est une "squelettisation". Celle-ci consiste à réduire progressivement la quantité de pixels des zones pertinentes, jusqu'à obtenir une représentation formée de lignes d'un seul pixel de large. Les propriétés topologiques de la forme d'origine ainsi que ses propriétés géométriques sont conservées. La méthode implémentée, issue de la libraire "Sikit-image" [252], provient des travaux de Zhang et. al. [253]. La squelettisation est alors effectuée par des passages successifs sur l'image, chacun supprimant les pixels en bordure de la structure sous la condition que la connectivité des objets n'est pas rompue. Le processus est itéré jusqu'à ce qu'aucun pixel ne puisse être supprimé sans respecter cette condition. Le lissage effectué lors de l'affinage de l'image seuillée permet de limiter l'apparition de "branches" courtes le long du squelette.

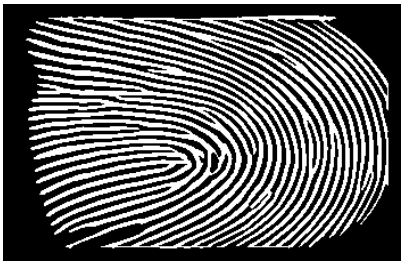


(a) Image seuillée et affinée - Moyennes

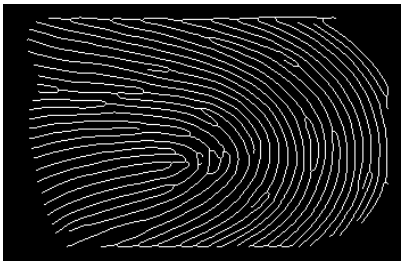


(b) Squelettisation du seuillage par moyennes

FIGURE 5.25 – Seuillage par moyennes locales et squelettisation de l'empreinte filtrée et corrigée

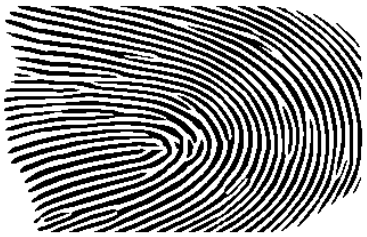


(a) Image seuillée et affinée - Niblack

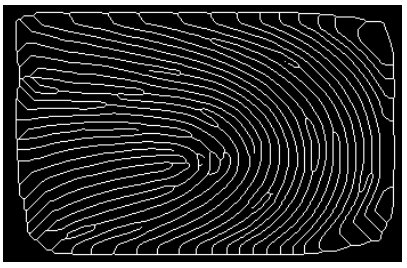


(b) Squelettisation du seuillage de Niblack

FIGURE 5.26 – Seuillage de Niblack et squelettisation de l'empreinte filtrée et corrigée



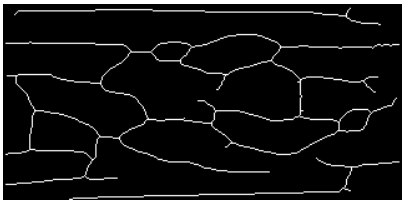
(a) Image seuillée et affinée - Otsu



(b) Squelettisation du seuillage d'Otsu

FIGURE 5.27 – Seuillage d'Otsu et squelettisation de l'empreinte filtrée et corrigée

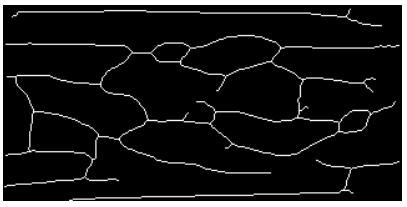
(a) Image seuillée et affinée - Moyennes



(b) Squelettisation du seuillage par moyennes

FIGURE 5.28 – Seuillage par moyennes locales et squelettisation du réseau vasculaire filtré et corrigé

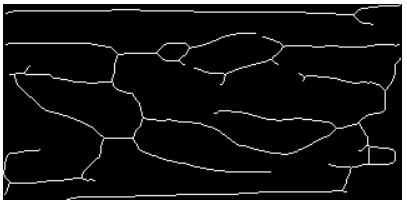
(a) Image seuillée et affinée - Niblack



(b) Squelettisation du seuillage de Niblack

FIGURE 5.29 – Seuillage de Niblack et squelettisation du réseau vasculaire filtré et corrigé

(a) Image seuillée et affinée - Otsu



(b) Squelettisation du seuillage d'Otsu

FIGURE 5.30 – Seuillage d'Otsu et squelettisation du réseau vasculaire filtré et corrigé

#### 5.2.2.3/ Extraction des caractéristiques

L'extraction de caractéristiques, décrite par les figures 5.31 et 5.32 est une étape décisive dans la mise en correspondance des images d'empreintes digitales et des images du réseau vasculaire. Une étape de détection des points d'intérêts commune à ces deux modalités est appliquée. Celles-ci présentant toutes deux des structures similaires, ce sont les fins de lignes et les bifurcations (ainsi que les croisements) qui sont détectées. Le premier niveau de précision des caractéristiques des empreintes digitales peut également être utilisé en complément des minuties du niveau 2.

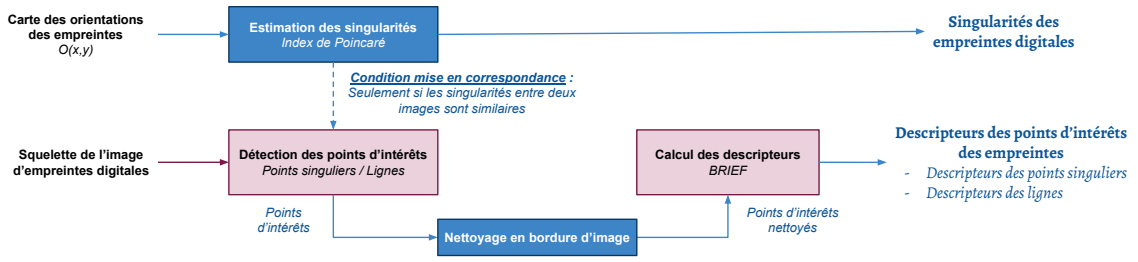



Figure 5.31 – Description du protocole de la phase d'extraction des caractéristiques des empreintes digitales


Figure 5.32 – Description du protocole de la phase d'extraction des caractéristiques des vaisseaux

**Détection des singularités *(Empreintes seulement)* :**
De par la composition du motif des empreintes digitales, il est possible d'extraire des informations à différents niveaux de précision, comme il a été montré au début de ce chapitre. Le premier niveau consiste à détecter l'aspect général de l'empreinte. Différentes structures, appelées singularités, peuvent être observées et leur combinaison permet de distinguer des groupes d'empreintes digitales. Leur traitement peut s'effectuer à partir de la carte des orientations locales, $O(x, y)$, calculée précédemment (lors du paramétrage du filtre de Gabor). Une des méthodes de la littérature, permettant cette détection de singularités locales et définissant des catégories d'empreintes selon les singularités détectées, repose sur l'indice de Poincaré [254]. Une présentation de cette méthode peut être trouvée en annexe B.2.1.3.

Cette détection de singularité représente un premier critère de comparaison des empreintes digitales, fiabilisant la mise en correspondance des minuties. En effet, effectuée directement après l'estimation des orientations, cette opération définit une condition nécessaire à la mise en correspondance des caractéristiques de niveau 2. En d'autres termes, si les deux images d'empreintes digitales comparées sont de mêmes types (déterminés par la table de référence de l'indice de Poincaré), alors le processus de détection des points d'intérêts peut être effectué. Dans le cas contraire, l'authentification de cette modalité est refusée. Les expérimentations préliminaires de cette méthode ne nous ayant pas fourni de résultats pertinents, nous ne l'intégrons pas à cette étude. Elle représente cependant une piste intéressante à considérer en perspectives de nos travaux.

### Détection des points caractéristiques *(vaisseaux et empreintes)* :

**Extraction des minuties :**
Cette extraction consiste à détecter, dans l'image squelettisée, les coordonnées des points d'intérêts formés par les fins de lignes, les bifurcations, les croisements de lignes, et les lignes elles mêmes. Dans le cas des empreintes digitales, nous ne cherchons pas à déterminer le motif formé des lignes autour de ces points d'intérêts. Comme pour les vaisseaux, seuls ces points sont pris en compte.

La recherche de ces points d'intérêt est effectuée par une nouvelle fenêtre glissante de taille $3 \times 3$, composées d'éléments de valeur 1 excepté l'élément central qui est à 0. Cette fenêtre est déplacée dans l'image et centrée à la position de chaque pixel des lignes de l'image squelettisée (chaque pixel différent de la valeur 0). Les pixels formant le squelette étant de valeur 1, la convolution de cette fenêtre à l'image permet de compter le nombre de pixels contenant de l'information dans le voisinage direct du pixel central de la fenêtre. Le résultat de cette convolution permet de catégoriser le type du point sur lequel est centrée la fenêtre, répartis sous quatre catégories. Une valeur à 1 indique que le pixel en question appartient à une ligne, une valeur égale à 2 définit une fin de ligne, une valeur de 3 distingue une bifurcation et enfin une valeur de 4 représente une jonction de lignes (ou croisement). La figure 5.33 illustre cette recherche de minuties pour chaque catégorie. Par souci de lisibilité, l'arrière plan est représenté en blanc, et les lignes sont représentées en gris. Les pixels de couleur représentent le pixel qui ont été assignés soit à une ligne (bleu), soit à une fin de ligne (rouge), soit à une bifurcation (vert) ou encore à une jonction (orange).

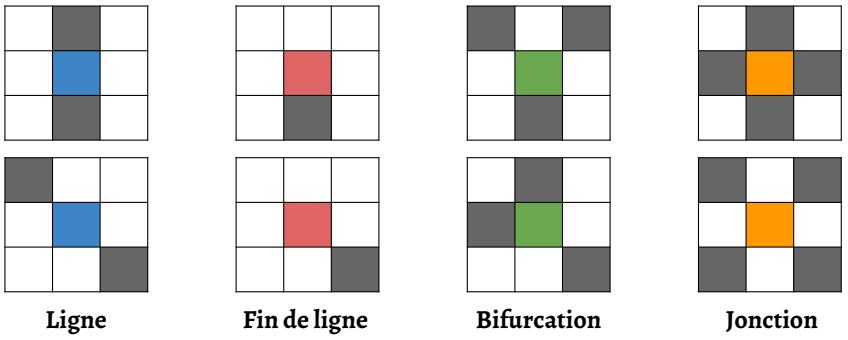



FIGURE 5.33 – Exemple de détection de minutie dans la fenêtre glissante. La couleur des pixels représente le type de point d'intérêt détecté

Un mauvais seuillage à l'emplacement d'une bifurcation ou d'une jonction produit un "trou" dans le squelette à cette position, conduisant à une détection de plusieurs fins de lignes (une pour chaque ligne formant la bifurcation ou la jonction). De plus, les jonctions sont généralement peu présentes dans les images. C'est pourquoi, dans la suite de ces travaux, nous regroupons ces trois types de minuties sous la même catégorie. En d'autres termes, nous retenons deux catégories de points d'intérêts : les lignes d'une part et les "points singuliers" d'autre part, composées des autres minuties (fin de ligne, bifurcation et jonction). Par ailleurs, considérer tous les pixels formant les lignes comme des points caractéristiques revient à conserver un nombre trop important d'éléments, vis à vis de nos contraintes en taille de stockage. De plus, un certain équilibre entre le nombre de points observés dans chaque catégorie est primordial afin de ne pas favoriser l'une ou l'autre des catégories dans le processus de mise correspondance. Par conséquent, nous limitons la quantité de points formant les lignes à un maximum correspondant au nombre de points de la seconde catégorie. Ainsi, bien que le nombre de détection puisse varier entre les images, le ratio du nombre de points dans chaque catégorie est constant. Nous limitons également à un maximum la quantité de points d'intérêt totale conservée par image, qui, dans la suite du document, est noté $\xi$.

Les empreintes digitales sont soumises à une étape supplémentaire dans ce processus d'extraction de caractéristiques. Cette dernière étape consiste à "nettoyer" les minuties formées par les fins de lignes en bordure de l'image. En effet, les lignes des empreintes digitales étant coupées par les bords de l'image, l'algorithme détecte des fins de lignes non pertinentes. Plutôt que de supprimer toute minutie détectée à une certaine distance des bordures, nous proposons une méthode permettant une plus grande robustesse à la position des empreintes dans l'image.
Nous considérons la ligne et la colonne de pixels de l'image sur laquelle se trouve un pixel correspondant à une fin de ligne, comme le montre l'image 5.34. Sur celles-ci, les pixels précédents l'emplacement du point caractéristique (à la position $(x, y)$) et les pixels suivants sont alors sommés pour obtenir quatre valeurs ($\Phi_1$ à $\Phi_4$). Comme l'illustre l'équation 5.14, où $W$ et $H$ représentent respectivement la largeur et la hauteur de l'image, et $P_f(i; x, y)$ la $i^{me}$ fin de ligne détectée, cette fin de ligne est supprimée si l'une de ces sommes est nulles. En effet, si tel est le cas, cela signifie que cette fin de ligne est suffisamment isolée pour pouvoir être supprimée (elle n'est pas inscrite dans l'empreinte digitale).

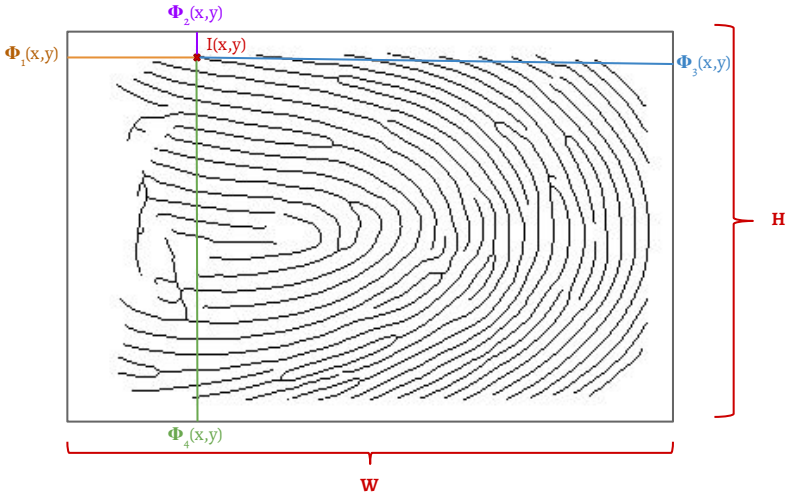
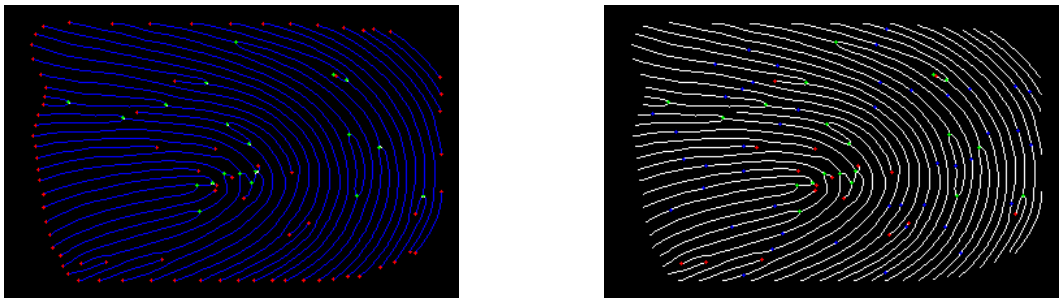



FIGURE 5.34 – Illustration du processus de nettoyage des minuties correspondant à une fin de ligne non inscrite dans l'empreinte digitale.

$$\Phi(x,y) = \left\{ \sum_{i=1}^{x-1} I(i,y) \;;\; \sum_{i=x+1}^{W} I(i,y) \;;\; \sum_{i=1}^{y-1} I(x,i) \;;\; \sum_{i=y+1}^{H} I(x,i) \right\} \tag{5.14}$$

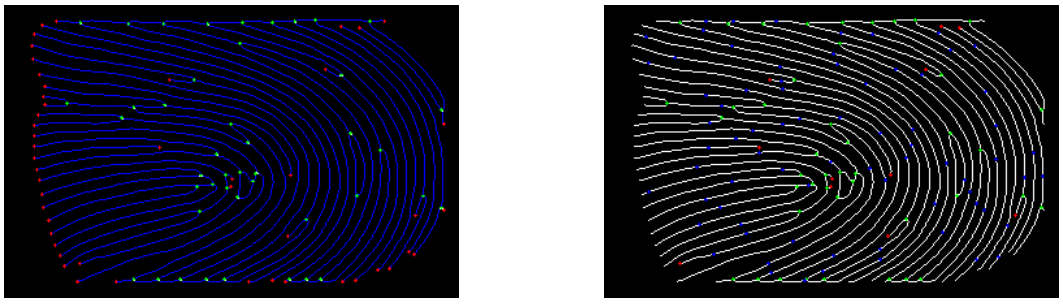
$$P_f(i; x, y) = 0, \quad \text{si } min[\Phi(x,y)] = 0$$

Les figures suivantes représentent en couleur les points d'intérêts détectés ainsi que ceux conservés après réduction du nombre de points des lignes et nettoyage des fins de lignes pour les empreintes. Les fins de lignes sont représentées par un point rouge, les lignes par un point bleu et l'ensemble des bifurcations et jonctions par un point vert. Nous rappelons que pour la correspondance, les fins de lignes sont ajoutées à l'ensemble des bifurcations et des jonctions.



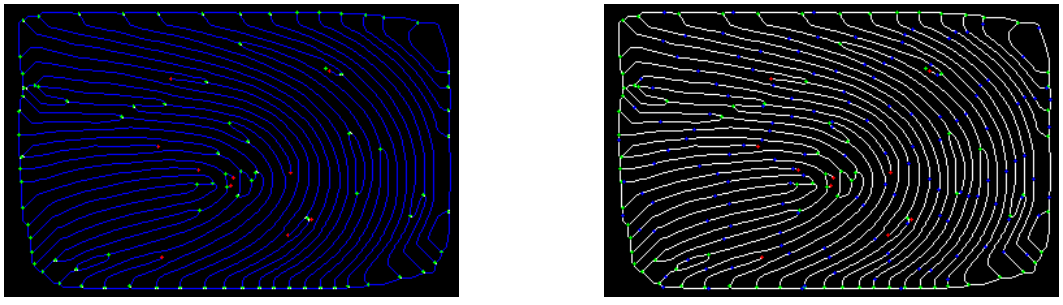
(a) Points d'intérêts détectés - Moyennes (b) Points d'intérêts conservés - Moyennes

FIGURE 5.35 – Points d'intérêts détectés et points d'intérêts conservés, à partir du squelette des empreintes digitales issu du seuillage par moyennes locales



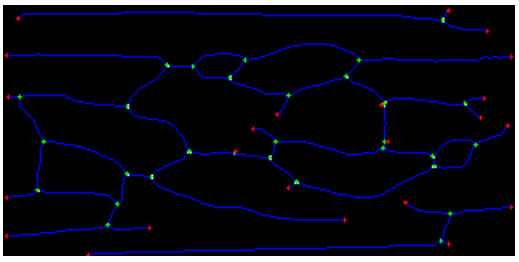
(a) Points d'intérêts détectés - Niblack (b) Points d'intérêts conservés - Niblack

FIGURE 5.36 – Points d'intérêts détectés et points d'intérêts conservés, à partir du squelette des empreintes digitales issu du seuillage de Niblack

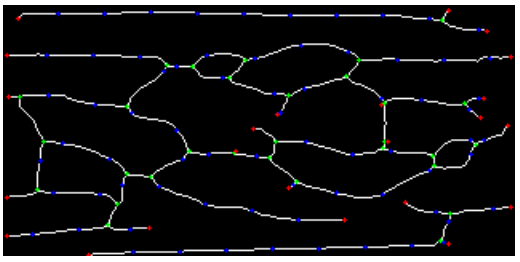


(a) Points d'intérêts détectés - Otsu (b) Points d'intérêts conservés - Otsu

FIGURE 5.37 – Points d'intérêts détectés et points d'intérêts conservés, à partir du squelette des empreintes digitales issu du seuillage d'Otsu

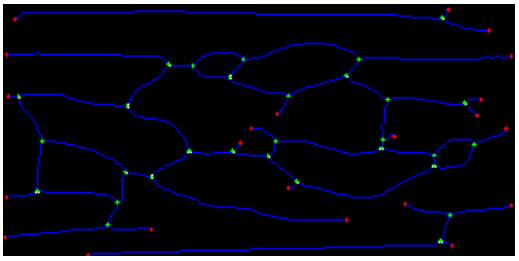


(a) Points d'intérêts détectés - Moyennes

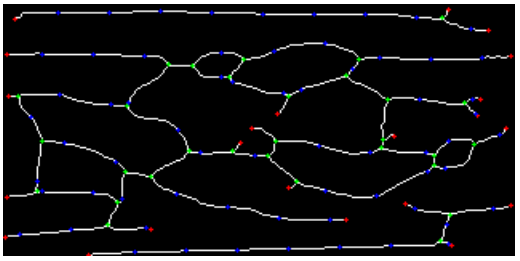


(b) Points d'intérêts conservés - Moyenens

FIGURE 5.38 – Points d'intérêts détectés et points d'intérêts conservés, à partir du squelette du réseau vasculaire issu du seuillage par moyennes locales

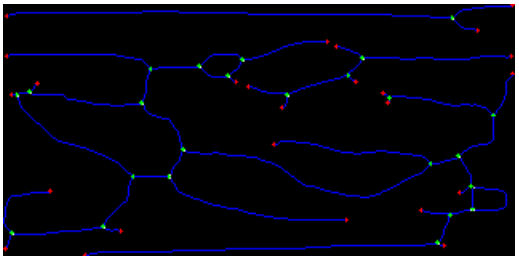


(a) Points d'intérêts détectés - Niblack

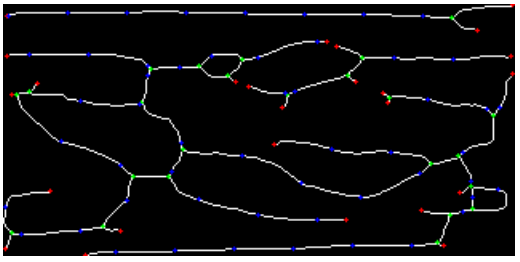


(b) Points d'intérêts conservés - Niblack

FIGURE 5.39 – Points d'intérêts détectés et points d'intérêts conservés, à partir du squelette du réseau vasculaire issu du seuillage de Niblack



(a) Points d'intérêts détectés - Otsu



(b) Points d'intérêts conservés - Otsu

FIGURE 5.40 – Points d'intérêts détectés et points d'intérêts conservés, à partir du squelette du réseau vasculaire issu du seuillage d'Otsu

**Création des descripteurs :**
La phase précédente permet de détecter la position des points d'intérêts que nous avons définis comme pertinents. Mais ces derniers ne permettent pas d'effectuer de correspondance directe entre deux images. Nous faisons alors usage d'un descripteur, permettant d'extraire des informations locales de l'image à l'emplacement de ces points d'intérêts. Étant donné que nous avons formé deux groupes de points d'intérêts (lignes et ensemble des fins de lignes, bifurcations et jonctions), nous créons deux matrices de description par image, chacune correspondant à un groupe de points d'intérêts. Nous avons sélectionné le descripteur BRIEF [255] (*Binary Robust Independent Elementary Features*) pour remplir ce rôle. Ce descripteur peut alors être appliqué soit à l'image améliorée en sortie du filtre de Gabor après correction ($E_C(x, y)$), soit à l'image squelettisée. Nous comparerons des résultats préliminaires sur ces deux images afin de sélectionner l'image la plus appropriée.

Le descripteur BRIEF est une alternative intéressante au descripteur SIFT (très utilisé dans la littérature). Sa complexité est en effet inférieure à celle de ce dernier, pour des performances de mise en correspondance relativement proches. Étant un descripteur assez simple, l'invariance de la description des points d'intérêts est limitée à la translation (pas d'invariance en en rotation ou en échelle). D'après Calonder et. al.[255], les performances sont seulement légèrement dégradées entre 0 et 15 degrés de rotation. Au delà, les performances subissent une perte conséquente. Cela suffit cependant à couvrir les majeures variations de positions présentes dans les bases de données utilisées.
La phase de description, basée sur la différence d'intensités de pixels sélectionnés aléatoirement parmi un voisinage autour du point d'intérêt (fenêtrage de $48 \times 48$ pixels centré sur le point d'intérêt), se divise en deux étapes. La zone observée est d'abord filtrée (flou gaussien) afin d'atténuer les erreurs dues au bruit. Par la suite, $n_d$ paires de pixels sont aléatoirement sélectionnés autour du point d'intérêt, suivant une distribution gaussienne centrée en ce point. Dans notre étude, 256 paires sont sélectionnées ($n_d = 256$). Pour chaque fenêtre (ou patch) $p$, les pixels de chaque paire (un pixel $a$ et un pixel $b$) sont comparés entre eux suivant la règle $\tau$ suivante :

$$\tau(p; a, b) = \begin{cases} 1 & si\ p(a) < p(b) \\ 0 & sinon \end{cases} \quad \Rightarrow \quad v_i(p) = \tau(p; a_i, b_i), \quad \forall i \in \{0, 1, ..., n_d\} \tag{5.15}$$

Pour chaque patch $p$ (associé à un point d'intérêt) et chaque point $i$ sélectionné dans $p$, cette comparaison donne lieu à un vecteur $v_i(p)$ de 256 éléments binaires ($n_d$ bits), composé des résultats de la règle $\tau$. Le vecteur de description de chaque point d'intérêt est ensuite divisé en mots de 8 bits, formant alors 32 éléments (256 bits / 8) compris alors entre 0 et 255. Le descripteur correspondant est appelé "BRIEF-k", où $k = n_d/8$. Dans notre cas, nous utilisons le descripteur "BRIEF-32". Pour chaque modalité de chaque personne, l'information utile stockée sur la carte sans contact est donc composée des matrices de description des lignes et des points singuliers (2 matrices au total), chacune formée d'autant de vecteurs de description que de points d'intérêts par catégorie.

#### 5.2.2.4/ Recherche de correspondances

Une fois les descripteurs obtenus pour chaque catégorie des deux images à comparer, une correspondance par "force brute" est effectuée ("*Brute Force Matching*"). Les descripteurs de chaque catégorie de points d'intérêts de la première image sont comparés avec les descripteurs de la catégorie associée dans la seconde image. Cette méthode calcule leur degré de correspondance par une mesure de distance. Pour chaque descripteur, la plus haute correspondance est donnée par la distance la plus courte. Différentes distances peuvent être utilisées, telles que les distance L1 ou L2 exposées dans le chapitre 3.

Le descripteur utilisé étant composé de caractéristiques binaires, nous utilisons la "distance de Hamming" pour estimer le degré de correspondance. Cette mesure de distance est obtenue facilement par l'opération d'une fonction "OU exclusif" ($A \oplus B$) entre deux descripteurs $A$ et $B$. Cette fonction produit alors un 1 si deux bits de $A$ et $B$ à une position $i$ sont différents, et un 0 s'ils sont égaux. La distance de Hamming entre deux descripteurs $A$ et $B$ de $n$ éléments (bits), $d_{H(A,B)}$, n'est autre que la somme des résultats de cette fonction pour chaque bit des deux descripteurs, tel que le montre l'équation 5.16.

$$d_H(A, B) = \sum_{i=1}^{n_d} (A_i \oplus B_i) \tag{5.16}$$

Plusieurs informations peuvent ensuite être extraites de cette comparaison. Tout d'abord, pour chaque catégorie, nous obtenons le nombre de correspondances entre les deux images, que nous pouvons comparer au nombre d'éléments formant les descripteurs. De plus, deux distances sont obtenues, une pour chaque catégorie de points, à savoir l'ensemble des points d'intérêts sur les lignes, et l'ensemble formé des fins de lignes, bifurcations et jonctions. Il est donc nécessaire d'effectuer une décision supplémentaire afin de valider l'authentification ou la rejeter.

Différentes méthodes peuvent être utilisées, la plus simple étant l'utilisation d'un seuil appliqué sur les distances obtenues pour les correspondances de chaque catégorie de points, de même qu'un seuil sur le nombre de correspondances de chaque catégorie. Cependant, la correspondance peut fournir de nombreuses autres informations. Nous proposons de calculer un certain nombre d'attributs supplémentaires, présentés par la table 5.1, et d'effectuer, pour chaque modalité (empreintes et réseau vasculaire), une sélection automatique des attributs les plus pertinents suivie d'une classification permettant de discriminer, de manière générale, les personnes autorisées et les intrus.
Il existe une large famille d'algorithmes sous-optimaux de sélection d'attributs. Nous avons retenu une variante de la méthode "*Sequential Backward Floating Selection*" (SBFS) [256] (présentée précédemment) qui consiste, en partant de l'ensemble des attributs, à retirer celui dont l'absence dans le sous-ensemble de l'itération suivante maximise les performances. Nous avons choisi d'utiliser l'erreur de classification globale comme fonction de coût à minimiser, maximisant alors les performances de classification. De la même manière que pour l'étude de l'authentification de visages, nous utilisons deux classifieurs, sur lesquels nous évaluons l'erreur globale lors de la SBFS, à savoir les SVM et les RF. Après apprentissage, le modèle de classification obtenu est commun à toutes les personnes enrôlées. Par conséquent, comme pour l'espace de visage créé par l'ACP, il ne présente aucune information personnelle et peut être stocké localement sur l'unité de calcul. Nous ne nous intéresserons pas à la taille de stockage de ces modèles de classification.

Dans cette étude, nous appliquons la SBFS pour chaque modalité, selon les trois types de seuillages que nous souhaitons évaluer (moyennes locales, Niblack et Otsu). La SBFS utilise un jeu de données d'apprentissage ainsi qu'un autre jeu de données de tests permettant d'évaluer les performances du classifieur. Pour rappel, nous avons sélectionné 30 personnes différentes par modalités. Pour les empreintes digitales, chaque personne dispose de 12 échantillons et 6 pour le réseau vasculaire. Nous devons constituer deux ensembles (apprentissage et test), chacun divisés en deux sous ensembles comprenant deux classes. Comme pour les visages, ces classes représentent d'une part les comparaisons positives (une image de référence $r_i$ d'une personne $p_a$ comparée à une image $r_j$ de la même personne, $i \neq j$), et d'autre part les comparaisons négatives (une image de référence $r_i$ d'une personne $p_a$ comparée à une image $r_j$ d'une personne $p_b$, $a \neq b$).

Parmi les images d'une même personne, il est possible d'effectuer au maximum 66 correspondances pour les empreintes et 15 pour les vaisseaux, sans considérer de doublons. Le nombre de comparaisons négatives possibles est bien plus important. Afin conserver un équilibre dans les classes, pour chaque personne de la base de donnée, nous limitons le nombre de correspondances à respectivement 66 et 15 images parmi les échantillons d'autre personnes, pour les empreintes et les vaisseaux.
Pour l'apprentissage, les personnes de références sont formées des 15 premières personnes de notre base de données. Pour le test, les personnes de références sont formées des 15 personnes suivantes. Chaque classe est alors composée de 225 éléments pour les vaisseaux (15 comparaisons x 15 personnes) et 990 pour les empreintes (66 comparaisons x 15 personnes). Afin de vérifier la robustesse aux intrusions, les classifieurs conservés (correspondant au jeu d'attributs fournissant l'erreur globale la plus faible dans le SBFS) seront évalués une dernière fois en utilisant les nombreuses tentatives d'intrusions (comparaisons négatives) que nous n'avons pas considérées lors de l'apprentissage.

| Catégories | Nom | Description des attributs |
|---|---|---|
| **Lignes** | $C_L$ | Nombre de correspondances parmi les points d'intérêt des lignes |
| | $R_L$ | Ratio entre le nombre de correspondances des lignes et le nombre minimal de descripteurs dans les deux ensembles comparés |
| | $\overline{d_L}$ | Moyenne des distances des correspondances des lignes |
| | $\sigma_L$ | Écart type des distances de correspondance des lignes |
| **Points singuliers** | $C_P$ | Nombre de correspondances parmi les points d'intérêt des points singuliers (fin de lignes, bifurcations et jonction) |
| | $R_P$ | Ratio entre le nombre de correspondances des points singuliers et le nombre minimal de descripteurs dans les deux ensembles comparés |
| | $\overline{d_P}$ | Moyenne des distances des correspondances des points singuliers |
| | $\sigma_P$ | Écart type des distances de correspondance des points singuliers |
| **Général** | $S_d$ | Somme des distances des deux catégories |
| | $S_{\overline{d}}$ | Somme des distances moyennes des deux catégories |
| | $\overline{S_d}$ | Moyenne de la somme des distances des deux catégories |
| | $S_\sigma$ | Somme des écarts types des correspondances des deux catégories |
| | $R$ | Ratio entre le nombre de correspondances totales et le nombre minimal d'éléments dans les descripteurs de chaque catégorie |
| | $C$ | Nombre de correspondances totales |

TABLE 5.1 – Les 14 attributs, calculés à partir de la correspondance entre les descripteurs de deux images, et transmis à la SBFS pour la sélection des attributs les plus pertinents

### 5.2.3/ Mise en place de l'étude

#### 5.2.3.1/ Optimisation des paramètres des algorithmes

Du fait des nombreuses étapes de traitement dans la chaîne algorithmique, cette étude est très paramétrique. Parmi les nombreuses combinaisons possibles, seul un certain nombre peut fournir des résultats optimaux. Par soucis de rapidité de développement, nous les avons fixés de manière expérimentale. Cependant ce paramétrage peut être optimisé de manière automatique afin d'augmenter la robustesse des prétraitements. Cette phase d'optimisation pourra être inclue dans la continuité de ces travaux, comme décrit dans l'annexe B.2.2. Cette annexe présente également les paramètres fixés pour chaque méthode utilisée dans la chaîne de traitement.

#### 5.2.3.2/ Étude préliminaire

Comme énoncé précédemment, les descripteurs BRIEF sont calculés pour chaque image. Deux matrices de description sont alors créées, une pour les points d'intérêts des lignes, et une pour l'ensemble des fins de lignes, des bifurcations et des jonctions. Elles sont composées d'autant de "lignes" (vecteurs de description) que de points d'intérêts, chacune étant composée de 32 mots de 8 bits. En suivant le protocole de la SBFS décrit précédemment, ces descripteurs sont comparés deux à deux, fournissant un jeu de 14 attributs issus de calculs sur les informations fournies par cette correspondance. Deux classifieurs sont entraînés à partir de ces attributs, les SVM et les RF. Comme étude préliminaire, figure 5.41, nous comparons le taux d'erreur obtenu par la SBFS en fonction de l'image à laquelle est appliquée le descripteur, à savoir l'image squelettisée ou l'image corrigée en sortie du filtre de Gabor ($E_x(x, y)$). De plus, nous comparons trois méthodes de seuillage (Moyennes locales, Niblack et Otsu) afin d'observer leur influence quant à la stabilité de la détection des points d'intérêts. L'objectif ici est de déterminer l'image de description fournissant les meilleures performances, afin d'orienter les tests suivants.

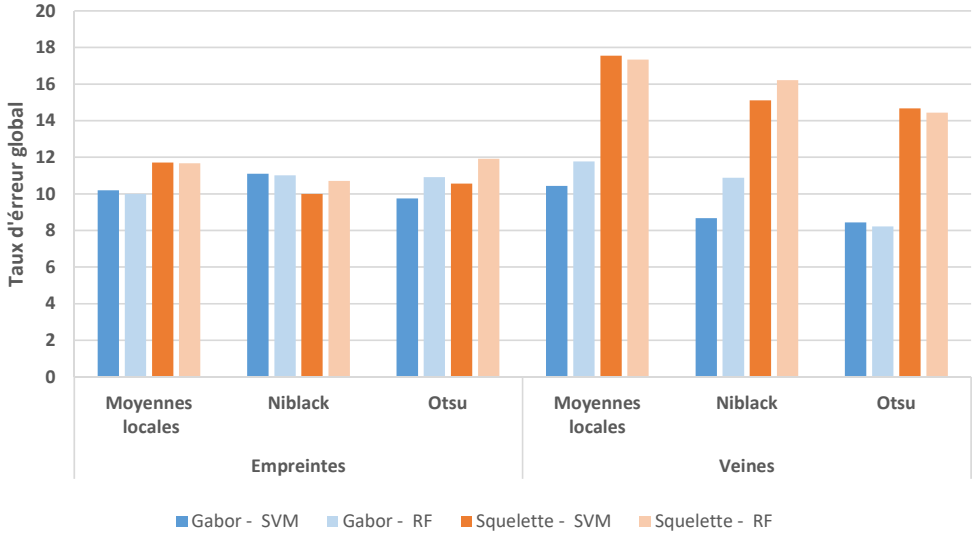



FIGURE 5.41 – Taux d'erreur global de la correspondance selon le type de seuillage, l'image à laquelle est appliquée le descripteur, et la SBFS effectuée par les SVM et les RF

D'une manière générale, on observe que l'erreur globale des SVM est proche des RF, quelle que soit la configuration. Pour les empreintes digitales, l'utilisation de l'image squelettisée pour la description des points d'intérêts produit des résultats similaires à l'utilisation de l'image corrigée en sortie du filtre de gabor ($E_C$). Cependant, pour les vaisseaux, l'erreur est plus importante (jusqu'à 18%) lorsque l'image squelittisée est utilisée comme source de description. Pour la suite de cette étude, nous utilisons l'image corrigée $E_C$ comme source de description. Celle-ci, pour les deux modalités du doigt, fournit un taux d'erreur global compris entre 8% et 12%, quelle que soit la configuration et le classifieur utilisé. La squelettisation est néanmoins toujours nécessaire, mais uniquement pour la détection des points d'intérêts.

## 5.3/ Résultats de la chaîne de traitements

Dans un premier temps, nous allons présenter les résultats d'authentification de chaque modalité pour les trois méthodes de seuillage sélectionnées. Les métriques d'évaluation des classifieurs SVM et RF sont comparées, pour les attributs résultants de la mise en correspondance des descripteurs issus des images corrigées en sortie du filtre de Gabor ($E_C$). Pour chaque configuration, les tailles de stockage des descripteurs à stocker sur la carte sans contact seront comparées. La configuration offrant les résultats optimums relativement à nos contraintes sera ensuite sélectionnée.
Dans un second temps, une étude d'adéquation algorithme et architecture sera effectuée afin de réduire la taille de stockage des descripteurs afin qu'ils puissent être stockés sur la carte individuelle. Deux méthodes seront évaluées et nous observons leur influence sur les performances relativement au gain de taille. D'une part, nous proposons de limiter le nombre de points d'intérêts conservés dans l'image, et d'autre part nous agissons directement sur le descripteur en réduisant sa précision pour chaque point décrit.

### 5.3.1/ Évaluation des performances d'authentification

Pour chaque configuration traitée, les table 5.2 et 5.3 présentent les métriques d'évaluation de la classification, pour respectivement les empreintes et les vaisseaux. Nous représentons la sensibilité (Se.), la spécificité (Sp.), la précision (Pr.) et la justesse (Ju.) des mesures. Les tailles de stockage moyennes ($\overline{T}$) et maximales ($T_{max}$) des deux ensembles de descripteurs (lignes et points singuliers), en kilooctets, sont également considérées. Du fait du type de données composant les descripteurs, ces deux ensembles sont stockés au format binaire, minimisant ainsi la taille de stockage. Ces résultats correspondent à la moyenne des 990 correspondances de tests pour les empreintes et des 225 correspondances de tests pour les vaisseaux sanguins.

On remarque que les deux classifieurs employés fournissent des résultats très restrictifs, quelle que soit la modalité. La spécificité est supérieure à 98%, et atteint même les 100% pour certaines configurations du réseau vasculaire. En d'autres termes, ces résultats traduisent un très faible taux de faux positifs. De même, la précision, exprimant la quantité de faux positifs relativement aux nombre d'échantillons classés positivement, est proche de la spécificité. La principale source d'erreurs vient de la sensibilité et donc du nombre de faux négatifs. La sensibilité se situe en effet aux alentours de 80% quel que soit le type de seuillage, la modalité et le classifieur. Cependant, l'objectif étant la définition d'un système de sécurité restrictif, ces résultats démontrent que nos classifieurs respectent nos contraintes. De plus, la justesse d'authentification oscille dans une plage comprise entre 88% et 92%. Par conséquent, nous pouvons conclure qu'avec les attributs sélectionnés par la SBFS, le type de seuillage et le type de classifieur ont peu d'incidence sur les performances de classification.

| Type de seuillage | SVM [%] | | | | RF [%] | | | | Desc. [Ko] | |
|---|---|---|---|---|---|---|---|---|---|---|
| | Se. | Sp. | Pr. | Ju. | Se. | Sp. | Pr. | Ju. | $\overline{T}$ | $T_{max}$ |
| Moyennes locales | 81,1 | 98,5 | 98,2 | 89,9 | 80,8 | 99,2 | 99,1 | 90,0 | 3,2 | 8,6 |
| Niblack | 79,2 | 98,6 | 98,3 | 88,9 | 80,0 | 98,0 | 97,5 | 89,0 | 4,1 | 8,6 |
| Otsu | 81,8 | 98,7 | 98,4 | 90,3 | 80,3 | 97,9 | 97,4 | 89,1 | 6,7 | 12,9 |

TABLE 5.2 – Sensibilité (Se.), spécificité (Sp.), précision (Pr.) et justesse (Ju.) moyennes (en pourcent) de la correspondance des **empreintes digitales** avec le descripteur **BRIEF-32** appliqué aux images corrigées en sortie du filtre de Gabor ($E_C(x,y)$), selon la décision des SVM et des RF, et leur tailles de stockage moyennes ($\overline{T}$) et maximales ($T_{max}$) en kilooctets

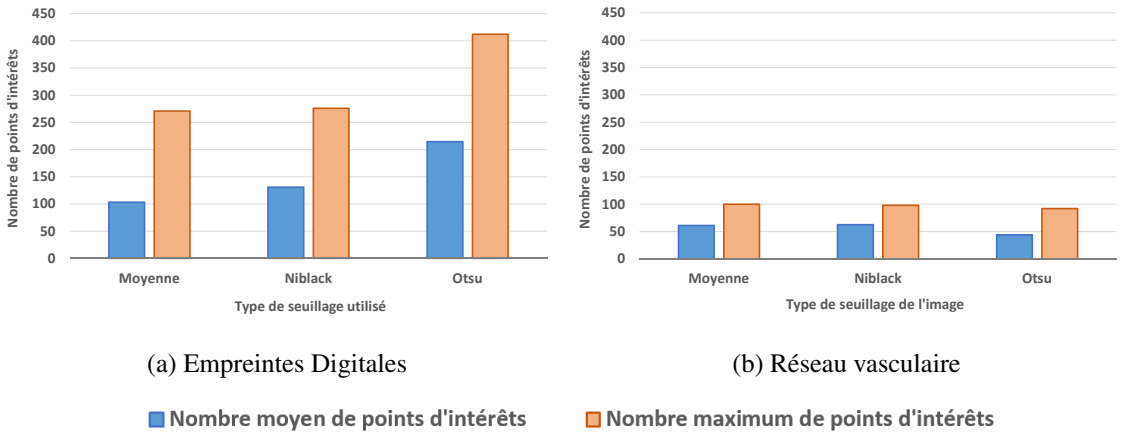
| Type de seuillage | SVM [%] | | | | RF [%] | | | | Desc. [Ko] | |
|---|---|---|---|---|---|---|---|---|---|---|
| | Se. | Sp. | Pr. | Ju. | Se. | Sp. | Pr. | Ju. | $\overline{T}$ | $T_{max}$ |
| Moyennes locales | 81,3 | 97,8 | 98,3 | 89,6 | 77,3 | 99,1 | 98,9 | 88,2 | 1,9 | 3,1 |
| Niblack | 84,4 | 98,2 | 97,9 | 91,3 | 78,2 | 100 | 100 | 89,1 | 2,0 | 3,1 |
| Otsu | 83,1 | 100 | 100 | 91,6 | 84,4 | 99,1 | 99,0 | 91,8 | 1,4 | 2,9 |

TABLE 5.3 – Sensibilité (Se.), spécificité (Sp.), précision (Pr.) et justesse (Ju.) moyennes (en pourcent) de la correspondance des **vaisseaux** avec le descripteur **BRIEF-32** appliqué aux images corrigées en sortie du filtre de Gabor ($E_C(x,y)$), selon la décision des SVM et des RF, et leur tailles de stockage moyennes ($\overline{T}$) et maximales ($T_{max}$) en kilooctets

L'objectif principal de ces travaux de thèse est d'établir une authentification biométrique par fusion de plusieurs modalités biométriques. Afin de se conforter aux recommandations en vigueur quant à la protection de la vie privée, toute information personnelle doit être stockée sur un support personnel. Par conséquent, vis à vis de notre étude, le classifieur propre au visage du sujet, ainsi que les descripteurs des deux modalités de ce chapitre doivent cohabiter sur le même support. Or, ce support impose une forte contrainte de taille de stockage. La condition sur la taille de stockage de chaque élément, jusque là fixée à 8 Ko, est donc à ajuster en fonction de chaque élément à stocker. À des fins de simplification, nous définissons la taille maximale de chaque élément comme étant un tiers des capacités de stockage de la carte, soit une limite de 2,6 kilo-octets.

Comparé au seuillage par moyennes locales, le seuillage de Niblack appliqué aux empreintes digitales fourni des ensembles de descripteurs dont la taille de stockage moyenne est supérieure. De même, les tailles de stockage moyennes et maximales des ensembles de descripteurs issus du seuillage d'Otsu sont supérieures à celles des descripteurs issus du seuillage de Niblack. Cela s'explique par l'observation des images squelettisées après ce seuillage. Ces méthodes, et principalement le seuillage d'Otsu, produisent en effet un "contour" autour squelette de l'empreinte, donnant alors lieu à un grand nombre de bifurcations. En comparaison, avec le seuillage par moyennes locales, l'emplacement de ces bifurcations se trouve être des fins de lignes qui ont été supprimées par l'opération de nettoyage des point d'intérêts que nous avons mis en place. Une opération similaire, supprimant les bifurcations en bordure du squelette, peut être envisagée pour le seuillage des empreintes digitales par la méthode d'Otsu.

Cette tendance est inversée pour le réseau vasculaire. Les seuillages de ces images par moyennes locales et par la méthode de Niblack produisent un nombre similaire de points d'intérêts. Par conséquent, la taille de stockage des descripteurs varie peu. Le seuillage d'Otsu en revanche, bien que présentant des résultats d'authentification plus élevés, est visuellement moins adapté à ce type de modalité. En effet, les images intermédiaires présentées précédemment mettent en évidence un seuillage plus grossier des vaisseaux. Le nombre de points d'intérêts détectés est par conséquent réduit. Comme nous fixons le nombre de points d'intérêts des lignes à la même valeur que le nombre de points de la seconde catégorie, la taille de stockage requise pour ces descripteurs diminue. Pour illustrer les différences de tailles de stockage entre les ensembles de descripteurs pour chaque configuration, la figure 5.42 expose le nombre de points d'intérêts moyen et maximum détectés dans les images pour chaque type de seuillage.




(a) Empreintes Digitales (b) Réseau vasculaire

FIGURE 5.42 – Nombre de points d'intérêts moyens et maximum parmi les images d'empreintes digitales et de réseaux vasculaires, selon le type de seuillage d'image

À ce stade, seuls les descripteurs des vaisseaux sanguins correspondent à nos nouvelles contraintes de taille de stockage. Afin de stocker les ensembles de descripteurs des empreintes digitales et des vaisseaux sur la carte sans contact, il est nécessaire d'intervenir sur la composition même de ces derniers. Nous proposons donc d'effectuer une étude d'adéquation algorithme et architecture dans le but de réduire la taille de stockage des descripteurs.
Cette étude a pour but d'établir une tendance quant aux variations de performances relativement à la réduction de la taille de stockage des descripteurs. Par conséquent nous restreignons cette étude à une configuration. Les performances d'authentification étant relativement proches entre les différents types de seuillage d'image, nous proposons d'utiliser le seuillage par moyennes locales associé à une classification des attributs issus de la correspondance par les SVM. Les descripteurs issus de ce seuillage étant déjà proche de la limite que nous imposons, nous pouvons estimer que la perte probable de performances sera limitée.

### 5.3.2/ Adéquation algorithme et architecture

#### Description des approches

Il existe différentes manières de procéder pour la réduction de la taille de stockage de nos ensembles de descripteurs. Dans la suite de ces travaux, nous confrontons deux de ces méthodes.

**Réduction du nombre de points d'intérêts - Descripteurs de 256 bits (BRIEF-32) :**
D'une part, nous proposons de réduire le nombre de descripteurs par image, et donc de tronquer le nombre de points d'intérêts. Le descripteur étant stocké au format binaire, sa taille de stockage dépend directement de la quantité de point d'intérêts décrits. Soit une taille de stockage limite $T_{max}$ en kilooctets et une précision $k$ du descripteur, l'équation suivante définit les nombres maximums de points d'intérêt à conserver pour une image, $\xi$, et par catégorie, $\xi_C$ (où $C \in \{L, P\}$ avec $L$ la catégorie des lignes et P les points singuliers) :

$$\begin{aligned} \xi &= 1024 \frac{T_{max}}{k} \\ \xi_C &= \frac{1}{2}\xi, \quad \text{où } C \in \{L, P\} \end{aligned} \tag{5.17}$$

En imposant une taille maximale de 2,6 Ko pour la description d'une image, et en prenant en compte une taille de 32 octets par descripteurs (BRIEF-32), on en déduit qu'il faut au maximum 83 points d'intérêt par image. Nous définissions cette limite à $\xi = 80$ points d'intérêts afin de conserver $\xi_C = 40$ points par catégorie et de conserver une certaine marge sur la taille de stockage. La taille maximale d'un ensemble de descripteurs pour une image est alors de 2,5 Ko.

**Réduction du la précision des descripteurs - Descripteurs de 128 bits (BRIEF-16) :**
D'autre part, la taille d'un ensemble de description peut être réduite en diminuant le nombre de paires de pixels comparés dans la création d'un descripteur, diminuant donc le nombre de bits utilisés pour sa représentation. Le descripteur utilisé est composé de 256 bits, soit 32 octets. Nous proposons alors d'observer l'influence d'une représentation de 128 bits, soit 16 octets (BRIEF-16). Cela permet alors de réduire de moitié la taille de stockage, tout en conservant le même nombre de points d'intérêts. Étant donné que les valeurs mêmes des descripteurs sont modifiées par cette réduction de précision, il est nécessaire d'effectuer une nouvelle SBFS avec les SVM.
Cette méthode étant confrontée à la même problématique que précédemment, une restriction du nombre de points d'intérêts peut s'avérer nécessaire. À partir de l'équation 5.17 et avec cette nouvelle représentation des descripteurs, nous en déduisons un maximum de 166 points d'intérêts par image. De la même manière que précédemment, nous arrondissons cette valeur à $\xi = 160$ (soit $\xi_C = 80$, permettant de conserver une marge sur cette taille de stockage et de maintenir un facteur deux entre les paramètres des deux cas étudiés. En effet, pour une précision $k$ des descripteurs deux fois plus petite (passant de 32 à 16), nous conservons le double de point d'intérêts $\xi$ par image (passant de 80 à 160).

#### Performances d'authentification

Ici, nous observons l'influence des deux méthodes de réduction de la taille de stockage des descripteurs sur les performances d'authentification. Pour les descripteurs 256 bits, la limite $\xi_C$ est fixée à 40 points d'intérêts. Cette limite est fixée à 80 points d'intérêts pour les descripteurs 128 bits. Dans le but d'observer les limitations d'une telle procédure, nous étudions deux limites de la quantité de points d'intérêts conservés pour chacune des deux méthodes proposées. Ainsi, nous observons deux limites supplémentaires, correspondant à une nouvelle division par deux de la taille de stockage, à savoir 20 et 40 points d'intérêts pour respectivement les descripteurs 256 bits et les descripteurs 128 bits.

Les table 5.4 et 5.5 regroupent les résultats de ces expérimentations, pour respectivement les descripteurs BRIEF-32 et BRIEF-16. Ces résultats sont présentés pour les deux $\xi_C$ étudiés de chaque descripteurs, ainsi que pour la configuration précédente où aucun seuil n'est appliqué sur ces points d'intérêts (lignes "N.A"). Quatre métriques d'évaluation de classification sont utilisées, à savoir la sensibilité (colonne "Se."), la spécificité (colonne "Sp."), la précision (colonne "Pr.") et la justesse (colonne "Ju."). De plus, pour chaque évaluation, cette table présente la taille de stockage moyenne des ensembles de descripteurs à stocker sur la carte ($\overline{T}$), ainsi que la taille maximale parmi tous les ensembles calculés ($T_{max}$).

| Modalité | $\xi_C$ | Se. | Sp. | Pr. | Ju. | $\overline{T}$ | $T_{max}$ |
|---|---|---|---|---|---|---|---|
| **Empreintes** | **N.A** | 81,1 % | 98,5 % | 98,2 % | 89,8 % | 3,2 Ko | 8,5 Ko |
| | **40** | 67,0 % | 99,2 % | 98,8 % | 83,1 % | 2,4 Ko | 2,5 Ko |
| | **20** | 62,1 % | 76,2 % | 72,3 % | 69,1 % | 1,3 Ko | 1,3 Ko |
| **Vaisseaux** | **N.A** | 81,3 % | 97,8 % | 97,3 % | 89,6 % | 1,9 Ko | 3,1 Ko |
| | **40** | 81,3 % | 97,3 % | 96,8 % | 89,3 % | 1,8 Ko | 2,5 Ko |
| | **20** | 75,1 % | 99,1 % | 98,8 % | 87,1 % | 1,2 Ko | 1,3 Ko |

TABLE 5.4 – Sensibilité (Se.), spécificité (Sp.), précision (Pr.) et justesse (Ju.) moyennes, ainsi que tailles moyennes ($\overline{T}$) et maximales (T. max.) des descripteurs **BRIEF-32**. Points d'intérêts limités à $\xi_C$ et issus du seuillage par moyennes locales. Descripteurs appliqués aux images corrigées en sortie du filtre de Gabor ($E_C(x,y)$) et décision par SVM

| Modalité | $\xi_C$ | Se. | Sp. | Pr. | Ju. | $\overline{T}$ | $T_{max}$ |
|---|---|---|---|---|---|---|---|
| **Empreintes** | **N.A** | 79,0 % | 98,7 % | 98,4 % | 88,8 % | 1,6 Ko | 4,2 Ko |
| | **80** | 77,9 % | 98,9 % | 98,6 % | 88,4 % | 1,6 Ko | 2,5 Ko |
| | **40** | 61,3 % | 99,7 % | 99,5 % | 80,5 % | 1,2 Ko | 1,3 Ko |
| **Vaisseaux** | **N.A** | 76,4 % | 99,6 % | 99,4 % | 88,0 % | 1,0 Ko | 1,6 Ko |
| | **80** | 76,4 % | 99,6 % | 99,4 % | 88,0 % | 1,0 Ko | 1,6 Ko |
| | **40** | 76,4 % | 99,6 % | 99,4 % | 88,0 % | 0,9 Ko | 1,3 Ko |

TABLE 5.5 – Sensibilité (Se.), spécificité (Sp.), précision (Pr.) et justesse (Ju.) moyennes, ainsi que tailles moyennes ($\overline{T}$) et maximales (T. max.) des descripteurs **BRIEF-16**. Points d'intérêts limités à $\xi_C$ et issus du seuillage par moyennes locales. Descripteurs appliqués aux images corrigées en sortie du filtre de Gabor ($E_C(x,y)$) et décision par SVM

En comparant directement ces deux tableaux pour chaque modalité sans contrainte sur le nombre de points d'intérêts (ligne *NA*), on observe des résultats quasi similaires. Réduire la précision des descripteurs diminue légèrement la spécificité tout en augmentant légèrement la spécificité. Cependant, même sans limite sur le nombre de points d'intérêt, l'utilisation des descripteurs 128 bits produit une taille de stockage deux fois plus petite tout en conservant donc sensiblement les mêmes performances. Pour les deux procédés, excepté pour le réseau vasculaire associé au descripteurs BRIEF-16, il faut fixer une première limite $\xi_C$ afin que les tailles de stockage des ensembles de descripteurs répondent à nos contraintes d'espace de stockage.

La première limite $\xi_C$ observée dans ces deux tables permet de restreindre les tailles de stockage jusqu'à un maximum de 2,5 Ko. Pour le descripteurs 256 bits, la sensibilité des empreintes digitales diminue de 14% tandis que la spécificité augmente légèrement. La taille moyenne des ensemble de descripteurs associés est très proche de la limite que nous imposons. Les points d'intérêts détectés dans les images de vaisseaux étant proches de la limite $\xi_C$, cette limite diminue légèrement les performances tout en réduisant subtilement la taille de stockage.
Pour les descripteurs 128 bits, le nombre de points d'intérêts conservé permettant de répondre à nos contraintes est supérieur. Cela permet d'atteindre une taille maximale de 2,5 Ko en réduisant très peu les performances d'authentification. On observe en effet une taille moyenne de 1,6 Ko pour les ensembles de descripteurs des empreintes digitales, avec une perte de sensibilité d'environ 1%. Le nombre moyen de points d'intérêts par catégorie des vaisseaux sanguins dans une image étant de l'ordre de 50, une limite à 80 points n'a aucune incidence.

Pour finir, nous observons un dernier seuil $\xi_C$ permettant de limiter la taille de stockage à 1,6 Ko pour la description d'une image. Avec les descripteurs BRIEF-32, cette nouvelle réduction diminue la spécificité des empreintes digitales de plus de 20%, et leur sensibilité d'environ 5%, faisant alors passer la justesse de 83% à 69%. Les vaisseaux sanguins subissent quant eux seulement une perte de 6% sur la sensibilité, la spécificité étant augmentée de 2%, conservant une justesse proche de 90%.
L'utilisation des descripteurs BRIEF-16 permet de limiter cette perte de performances d'authentification. En effet, avec une taille de stockage assez proche des descripteurs 256 bits, les empreintes digitales conservent une justesse supérieure à 80%. Les fausses acceptations sont quasi nulles avec une spécificité proche de 100%, tandis que la sensibilité perd 16% pour atteindre presque 60%. Une nouvelle fois, le réseau vasculaire est peu impacté par cette limite $\xi_C$ associée aux descripteurs 128 bits. Cette limite étant toujours proche du nombre moyen de points d'intérêts détectés dans l'image, seule la taille de stockage moyenne est influencée.

L'étude de ces deux procédés nous permet de conclure quant à la méthode de réduction de taille de stockage à favoriser pour conserver un compromis intéressant entre performances d'authentification et espace de stockage nécessaire aux ensembles de descripteurs. Pour une même limite de taille de stockage, la comparaison des performances d'authentification de ces deux procédés montre qu'il est préférable d'utiliser des descripteurs moins précis, mais décrivant une image au travers de plus de points d'intérêts. En effet, la réduction de précision des descripteurs permet de conserver suffisamment de point d'intérêts pour entretenir une excellente restrictivité du système tout en maintenant un taux de faux rejets acceptable.

La table 5.6 présente les attributs sélectionnés par les SVM lors de la SBFS, pour les descripteurs 256 bits et 128 bits, décrivant les points d'intérêts détectés à partir de l'image issue du seuillage par moyennes locales. Les attributs en gras représentent, pour chaque modalité, les attributs qui ont sélectionnés par la SBFS avec les descripteurs BRIEF-32 et BRIEF-16. Cette table montre une certaine constance dans les attributs sélectionnés par la SBFS. Pour les empreintes digitales, quatre des cinq attributs sélectionnés pour les descripteurs 128 bits figurent parmi ceux retenus par la SBFS avec les descripteurs de 256 bits. Seul l'écart type des distances de correspondance des lignes ($\sigma_L$) a été ajoutée. Tous les attributs des vaisseaux sanguins sélectionnés pour les descripteurs 128 bits se trouvent parmi ceux sélectionnés pour les descripteurs 256 bits.

| Modalité | Attributs - BRIEF-32 | | | Attributs - BRIEF-16 | | |
|---|---|---|---|---|---|---|
| | Lignes | P. Singuliers | Général | Lignes | P. Singuliers | Général |
| **Empreintes** | $\boldsymbol{\overline{d_L}}$ | $\boldsymbol{C_P}$ ; $\boldsymbol{\overline{d_P}}$ ; $\sigma_P$ | $S_{\overline{d}}$ ; $S_\sigma$ ; $\boldsymbol{R}$ | $\boldsymbol{\overline{d_L}}$ ; $\sigma_L$ | $\boldsymbol{C_P}$ ; $\boldsymbol{\overline{d_P}}$ | $\boldsymbol{R}$ |
| **Vaisseaux** | $C_L$ ; $\sigma_L$ | $\overline{d_P}$ ; $\sigma_P$ | $\boldsymbol{\overline{S_d}}$ ; $\boldsymbol{S_\sigma}$ ; $\boldsymbol{R}$ | / | / | $\boldsymbol{\overline{S_d}}$ ; $\boldsymbol{S_\sigma}$ ; $\boldsymbol{R}$ |

TABLE 5.6 – Attributs (définis table 5.1) sélectionnés par le classifieur SVM avec la SBFS, donnant la plus faible erreur globale, avec les descripteurs **BRIEF-32** et **BRIEF-16**, appliqués au points d'intérêts détectés à partir des images seuillées par moyennes locales. Les attributs en gras représentent, pour chaque modalité, les attributs qui ont été sélectionnés pour les deux descripteurs

### 5.3.3/ ÉVALUATION DE LA ROBUSTESSE AUX INTRUS

Afin d'équilibrer les classes dans nos tests, relativement à la faible quantité d'échantillons positifs disponibles, les mesures effectuées dans ce chapitre ne prennent en compte qu'un faible nombre d'échantillons d'intrus. En implantation industrielle (ou réelle), la proportion d'échantillons d'intrus potentiels par rapport au nombre d'échantillons positifs est bien plus élevée. Nous proposons alors de vérifier l'aptitude de notre méthode à rejeter les intrus. Pour chaque modalité, nous sélectionnons la configuration fournissant le meilleur compromis entre nos contraintes et les performances d'authentification.
Par conséquent, pour les empreintes digitales, nous sélectionnons le descripteurs BRIEF-16 associé à une limitation de 160 points d'intérêts par image (80 par catégorie). Pour les vaisseaux sanguins, nous sélectionnons le descripteurs BRIEF-32 associé à une limitation de 80 points d'intérêts par image (40 par catégorie). Pour rappel, ces points d'intérêts proviennent de la détection sur l'image squelettisée issue du seuillage par moyennes locales.
La table 5.7 présente les résultats de cette évaluation, effectuée à partir de 12744 tentatives d'intrusion pour les empreintes digitales et les vaisseaux sanguins. La base de données des empreintes digitales présentant plus d'images par individu, le nombre de tests de cette modalité a été limité afin de correspondre à la quantité de tests des vaisseaux sanguins. De plus, nous profitons de la grande quantité d'images d'empreintes digitales par personne afin d'évaluer la stabilité de ces mesures, avec 50976 tentatives d'intrusion. Les paires d'images mises en correspondances ne figurent bien évidemment pas dans l'ensemble d'apprentissage des SVM.

| Modalité | Descripteur | $\xi_C$ | Tentatives d'intrusion | Er. | Sp. |
|---|---|---|---|---|---|
| **Empreintes** | **128 bits** | **80** | **12744** | 1,78 | 98,22 |
| | | | **50976** | 1,85 | 98,15 |
| **Vaisseaux** | **256 bits** | **40** | **12744** | 0,59 | 99,41 |

TABLE 5.7 – Évaluation de la robustesse aux intrus, pour les descripteurs sélectionnés (associés à un SVM), avec l'erreur ("Er." ou taux de fausse acceptation) et la spécificité (Sp.), en pourcent

Ces dernières évaluations montrent un taux d'erreur (ou ici taux de faux positifs) inférieur à 2% pour les empreintes et inférieur à 1% pour les vaisseaux. Avec seulement des échantillons négatifs, la spécificité peut être calculée et donc comparée à nos tests précédents. Égale à 98,2%, la spécificité des empreintes digitales pour 12744 authentifications est très proche de celle des tests précédents pour la même configuration (98,9%, deuxième ligne de la table 5.5). Les résultats des métriques obtenues pour 50976 tests démontrent une très bonne stabilité du système.

L'évaluation des vaisseaux sanguins montrent une très bonne robustesse aux intrus avec un taux de faux positifs est inférieur à 1%. En comparaison avec les tests précédents pour la même configuration (deuxième ligne des empreintes, table 5.4), on observe qu'ici, sur plus d'échantillons de tests, la sensibilité est bien plus élevée. Cette spécificité a en effet gagné près de 2%, passant de 97,3% à 99,4%.

## 5.4/ Discussion

Dans ce chapitre, nous avons présenté les méthodes de notre chaîne de traitement permettant d'effectuer une authentification de personnes soit par ses empreintes digitales, soit par son réseau vasculaire du doigt. Cette chaîne algorithmique présente plusieurs étages, chacun ayant une fonction distincte permettant d'optimiser les résultats de l'étage suivant.
Une correction d'image est effectuée afin de faciliter le seuillage de la modalité. L'image binaire résultant de ce seuillage permet d'obtenir un "squelette" de la structure linéaire formant la modalité. C'est à partir de ce squelette que peuvent être détectées les particularités du motif des empreintes ou des vaisseaux, à savoir des fins de lignes, des bifurcations, des jonctions, et des points sur les lignes elles-mêmes. Dans l'image corrigée, à la position de ces points d'intérêts, un descripteur est utilisé afin d'extraire des informations locales (ou des caractéristiques), et une mise en correspondance est effectuée entre les descripteurs de deux images d'une même modalité. Pour finir, de nouveaux attributs sont calculés à partir des informations fournies par cette correspondance, et permettent, grâce à une classification binaire effectuée suite à l'apprentissage d'un SVM, de valider ou non l'authentification.
Une des contraintes, imposées par l'application sur laquelle repose ces travaux, porte sur la taille de stockage des descripteurs qui sont à stocker sur la carte personnelle sans contact. L'étude présentée dans ce chapitre est donc divisée en plusieurs partie. D'une part, nous avons évalué les performances de notre méthode sans aucune contrainte. D'autre part, nous avons cherché à optimiser la taille de stockage par divers procédés, afin de répondre à notre cahier des charges. Une nouvelle évaluation a donc été effectuée afin de valider ce procédé.

Pour chaque modalité, la méthode proposée présente un taux d'erreur globale d'environ 10%. Une forte capacité à rejeter les intrus est à noter, tandis que le système présente un taux de faux négatifs d'environ 20%, pour les deux modalités. Cependant, bien que déjà faibles, les tailles des descripteurs ne permettent pas leur stockage sur la carte sans contact. Les deux pistes explorées, à savoir une limitation du nombre de points d'intérêts et une réduction de la précision des descripteurs, permettent de résoudre ce problème. La table 5.8 résume les résultats obtenus pour les configurations retenues.

| **Mod.** | **Desc.** | $\xi_C$ | **Attributs** | **Se.** | **Sp.** | **Pr.** | **Ju.** | $\overline{T}$ |
|---|---|---|---|---|---|---|---|---|
| **E** | 128 | 80 | $\overline{d_L}\,;\sigma_L\,;C_P\,;\overline{d_P}\,;R$ | 77,9 | 98,9 | 98,6 | 88,4 | 1,6 |
| **V** | 256 | 40 | $C_L\,;\sigma_L\,;\overline{d_P}\,;\sigma_P\,;\overline{S_d}\,;S_\sigma\,;R$ | 81,3 | 97,3 | 96,8 | 89,3 | 1,8 |

Table 5.8 – Tableau résumé des configurations retenues offrant le meilleur compromis entre performances (sensibilité, spécificité, précision et justesse en pourcent) et taille moyenne de stockage des descripteurs (en Ko), avec $T_{max} = 2,5Ko$

Les empreintes digitales présentent le meilleur compromis (performances / taille de stockage) avec les descripteurs 128 bits et une limite de 80 points d'intérêts par catégorie (160 par image), et le meilleur compromis pour les réseaux vasculaires est obtenu avec les descripteurs 256 bits et une limite de 40 points d'intérêts par catégorie (80 par image). Ainsi, les tailles de stockage moyennes des descripteurs de chacune de ces modalités sont inférieures à 2 Ko, avec un maximum de 2,5 Ko, ce qui reste inférieur à la limite fixée de 2,6 Ko pour chaque modalité. Les performances observées pour les configurations retenues présentent une excellente robustesse aux intrus. Après vérification sur un grand nombre de mises en correspondances d'images de personnes différentes, le taux de faux positifs des empreintes digitales est inférieur à 2% et celui ci est inférieur à 1% pour le réseau vasculaire. Cependant, comme précisé précédemment, le taux de rejet parmi les authentifications de personnes autorisées (taux de faux négatifs) est de l'ordre de 20% pour les deux modalités. Ces résultats, bien qu'assez faibles relativement au taux de faux positifs, conviennent aux besoins de l'application de ces travaux.

L'objectif principal étant de sécuriser l'accès à une zone de stockage de produits sensibles, notre système se doit d'être doté d'une forte robustesse aux intrus. Afin d'augmenter encore cette robustesse, et de pallier aux lacunes d'authentification des personnes autorisées, la section 6.1.2 du chapitre suivant traite du système de sécurité dans sa globalité, mettant en œuvre les trois modalités, traitées jusque là, par une fusion mutlimodale.

# 6

# Système de sécurité biométrique multimodal

Diverses modalités biométriques ont été exploitées dans les chapitres précédents. Elles ont été évaluées indépendamment les unes des autres et les protocoles de tests ainsi que les résultats y ont été décrits sans considérer de cible d'intégration précise. Ce dernier chapitre a pour objectif d'étudier le système de sécurité du projet dans sa globalité.

Dans un premier temps, il s'agit de présenter ce système et son fonctionnement. Les trois modalités traitées sont alors mises à contributions ensemble pour réaliser une authentification basée sur une fusion de décision. Cette authentification multimodale est alors plus robuste aux intrusions et aux usurpations d'identité qu'une authentification basée sur une unique modalité.
Dans un second temps, nous traitons des aspects d'intégration matérielle de ces méthodes au sein de caméras intelligentes. Celles-ci permettent un calcul interne afin de ne faire transiter sur le réseau que le résultat de l'authentification de chaque modalité. L'objectif est de vérifier la contrainte temps réel sur une cible à capacité plus réduite et à encombrement plus restreint qu'une plateforme PC (plateforme de développement).

Pour terminer, le prototype d'un système d'acquisition original des empreintes et des vaisseaux sanguins est présenté. Pensé comme un scanner rotatif, ce système permet d'acquérir ces deux modalités sur le tour du doigt. Ainsi, ce type d'acquisition est plus robuste aux potentielles variations de positions du doigt et l'usurpation (copie) est donc plus complexe.

## Sommaire

## 6.1/ Évaluation du système

Les méthodes étudiées précédemment ont pour vocation à être intégrées au sein d'un système industriel permettant de répondre à la problématique du projet, à savoir "sécuriser l'accès à une zone de stockage de produits sensibles". Un aperçu réalisé à partir d'une modélisation 3D, figure 6.1, illustre de manière concrète ce système en phase d'utilisation réelle. On y trouve une caméra frontale et une caméra latérale (caméras intelligentes ou "*Smart Camera*"), un lecteur de carte, un système d'acquisition original des modalités du doigt (que nous définirons dans ce chapitre). Une potentielle Interface Homme Machine (IHM) peut être ajoutée, permettant l'affichage de diverses informations et des consignes facilitant le processus d'authentification (telles que l'état du système, le résultat de la lecture de la carte, la marche à suivre, etc). Celle-ci peut également être dotée d'une fonctionnalité de signalement de dysfonctionnement ou de tout autre problème.

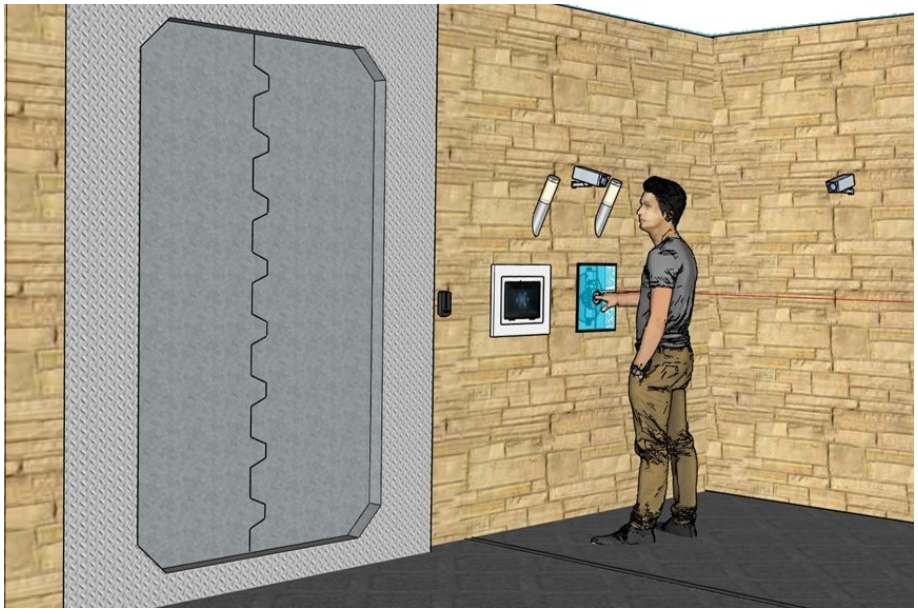


Figure 6.1 – Aperçu 3D du système

### 6.1.1/ Présentation du système complet

La première étape de la phase d'authentification est la lecture du badge sur le lecteur de carte. Celui-ci vérifie l'identité du badge afin de confirmer l'habilitation de la personne à accéder à la pièce de stockage. Conformément aux protocoles Nuc-Track, le lecteur interroge aussi un serveur dans le but de confirmer la raison de la présence de l'agent de par son agenda de missions. Si ces processus de vérifications sont validés, le lecteur déclenche le système des caméras intelligentes et transmet alors à ces dernières les données d'authentification lues dans la carte.

La caméra frontale (Smart Cam 1) se trouve donc être la caméra intelligente chargée d'effectuer l'authentification du visage de l'usager. Un système d'éclairage frontal permet de limiter les variations environnementales entre les images utilisées pour l'apprentissage et l'authentification. Cette caméra, détectant en premier lieu les visages, procède aux traitements biométriques seulement si un seul visage est détecté. Cette condition validée, la caméra intelligente transmet ensuite à l'unité de calculs principale le résultat de l'authentification avec le score de la prédiction. L'interface IHM peut potentiellement afficher des indications sur la position du sujet afin de garantir une orientation optimale du visage.

La fonction de la caméra intelligente latérale (Smart Cam 2) est complémentaire à la détection de personnes par la caméra frontale. Son unique rôle est de garantir l'unicité de passage en contrôlant qu'aucun autre individu ne se soit dissimulé pour entrer frauduleusement (grâce à une détection de silhouettes et/ou de visages). Dans un contexte de sécurité de ce type, il est en effet envisageable qu'une personne mal intentionnée tente de pénétrer dans l'espace de stockage, menaçant un agent habilité. Un sas d'accès à double portes peut être une solution alternative, classiquement utilisée pour contraindre l'unicité de passage sans recourt à la vision.

Afin de pallier aux possibles failles de sécurités induites par l'authentification du visage en 2D, nous avons proposé d'exploiter deux modalités supplémentaires, à savoir les empreintes digitales et un motif formé du réseau vasculaire du doigt. Ces deux modalités sont mesurées sur un seul doigt du sujet, ce qui permet de limiter le nombre de systèmes d'acquisitions à un unique dispositif. De plus, dans la suite de ce chapitre, nous proposons d'étudier un prototype d'acquisition original permettant de mesurer ces modalités par rotation d'une caméra à capteur linéaire autour du doigt.

Une unité de calcul principale contrôle l'ouverture de la porte et transmet également diverses informations au serveur et à l'IHM. Elle reçoit les résultats des traitements des caméras intelligentes et effectue la fusion de décisions sur la base des prédictions biométriques de chaque modalité. C'est cette décision globale d'authentification qui permet l'ouverture de la porte.

### 6.1.2/ Fusion de modalités biométrique

L'objectif est d'évaluer notre système de sécurité biométrique dans son fonctionnement industriel (ou réel). À l'aide des différentes évaluations réalisées jusque là pour chaque modalité, une sélection doit tout d'abord être faite quant à la configuration à retenir pour chacune d'elle. Par la suite, nous traitons des performances du système pour une fusion de modalités. Par soucis de rapidité de développement d'une preuve de concept, nous avons choisi d'utiliser la fusion par décision, décrite dans la section 2.2.1.2. Cette fusion est mise en œuvre de deux manières, à savoir un vote à la majorité et un vote à l'unanimité.
Tel que le décrit la figure 6.2, l'autorisation d'accès est donnée si et seulement si tous les classifieurs déterminent la personne comme étant autorisée pour le vote à l'unanimité, ou si au moins la moitié des classifieurs plus un acceptent les échantillons testés dans le cas du vote à la majorité. Ici, nous utilisons trois classifieurs. Par conséquent, pour le vote à la majorité, au moins deux classifieurs doivent définir la personne comme étant autorisée.

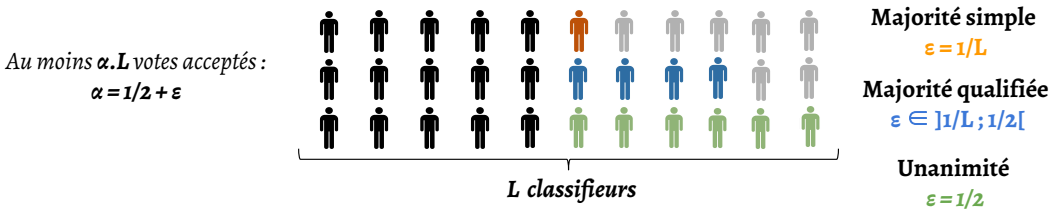



Figure 6.2 – Principe des votes à l'unanimité et à la majorité [257]

Dans cette section, la fusion est évaluée pour deux cas de figure distincts. D'une part, cette évaluation est effectuée le cas général où une personne se présente devant le système avec soit ses modalités valides (personne autorisée), soit toutes les modalités invalides (intrus). D'autre part, l'évaluation considère également le cas d'usurpation d'identité où un intrus se présente avec une modalité appartenant à une personne habilitée (cet intrus est donc un usurpateur).

#### 6.1.2.1/ Choix des configurations du système

Au vu des expérimentations présentées dans ce manuscrit et des contraintes du projet, nous pouvons sélectionner, pour chaque modalité, la configuration optimale offrant le meilleur compromis vis à vis de notre cahier des charges. Ces configurations sont exposées dans la table 6.1. Nous avions défini un espace de stockage de 2,6 Ko pour chaque modalité sur la carte sans contact. Sans considérer les temps de calculs, nous pouvons donc éliminer notre implémentation de l'authentification de visages basée sur le Deep Learning, laquelle requiert un espace de stockage beaucoup trop important. Parmi les configurations des méthodes de Machine Learning traditionnel disponibles dans la table 4.6, une seule produit des classifieurs personnels inférieurs à 2,6 Ko. Cette configuration met en œuvre le filtre de TanTriggs, avec un $\beta$ de 20%, associé à l'ACP seuillé à 50% et un classifieur SVM. La taille de stockage moyenne de ce modèle de classification est de 1,6 Ko. Les configurations retenues des empreintes digitales et des vaisseaux sanguins restent les mêmes que celles sélectionnées pour l'évaluation de la robustesse aux intrus sur un grand nombre de tentatives d'intrusion, dans le chapitre précédent.

| Modalité | Configuration sélectionnée | Se | Sp | Pr | Ju | $\overline{T}$ | $T_{max}$ |
|---|---|---|---|---|---|---|---|
| **Visage** | TanTriggs / $\beta$ = 20 / ACP 50 / SVM | 81,6 | 91,6 | 90,7 | 86,6 | 1,9 | 2,6 |
| **Empreintes** | BRIEF-16 / $\xi_C$ = 80 / SVM | 77,9 | 98,9 | 98,6 | 88,4 | 1,6 | 2,5 |
| **Vaisseaux** | BRIEF-32 / $\xi_C$ = 40 / SVM | 81,3 | 97,3 | 96,8 | 89,3 | 1,8 | 2,5 |

Table 6.1 – Performances des configurations retenues de chaque modalité, pour l'authentification biométrique par fusion de décision

L'erreur globale de chacune de ces modalités est de l'ordre de 12% (justesse d'authentification entre 86,6% et 89,3%) pour une taille de stockage des données biométriques inférieure à 2,6 Ko. Ces modalités présentent une bonne robustesse aux intrus avec moins de 3% de faux positifs pour les empreintes et les vaisseaux, et moins de 9% pour le visage. Cependant, la quantité de faux négatif est plus élevée avec 80% de sensibilité en moyenne sur ces trois modalités.

#### 6.1.2.2/ Système biométrique multimodal - Fusion de données biométrique

Nous évaluons d'une part le système de manière globale, avec des tentatives d'authentifications de personnes habilitées et des tentatives d'intrusion. Une tentative d'intrusion est définie comme une présentation de trois modalités appartenant à une même personne, différente de la personne de référence (c'est à dire des modalités différentes des références présentes sur la carte sans contact). D'autre part, nous évaluons la robustesse du système aux tentatives d'usurpation d'identité. Nous caractérisons cette usurpation comme une "copie" d'une des trois modalités d'une personne habilitée. Les deux autres sont alors différentes des modalités de l'individu de référence. Par conséquent, pour une personne de référence, une des trois modalités utilisées lui appartient (visage, empreintes ou vaisseaux), et les autres appartiennent à l'usurpateur. Nous comparons alors les résultats d'authentification pour une usurpation de chaque modalité. Nous proposons tout d'abord d'estimer les performances de la fusion de décision à partir des résultats généraux, rappelés par la table 6.1. Par la suite, nous validerons ces résultats expérimentalement. Une nouvelle base de donnée est construite, composée de 11 individus, chacun disposant des trois modalités traitées.

### Estimation des performances de fusion

Pour réaliser cette estimation, nous nous basons sur les métriques d'évaluations obtenues dans les chapitres précédents. Ces métriques représentent alors des probabilités d'évènements. Dans cette section, nous utilisons les notations utilisées par Kuncheva [257]. Ainsi, la sensibilité $U$ représente la probabilité d'une prédiction correcte (autorisation) parmi les échantillons de personnes autorisées. La spécificité $V$ représente la probabilité d'une prédiction correcte (refus) parmi les échantillons d'intrus. Soit $T$ un résultat positif et $A$ défini une personne autorisée. Alors la sensibilité et la spécificité représentent respectivement $U = P(T|A)$ et $V = P(\overline{T}|\overline{A})$. Par conséquent, $[1 - U]$, ou $P(\overline{T}|A)$, représente la probabilité d'obtenir un faux négatif parmi les échantillons de personnes autorisées (personne autorisée classée comme intrus) et $[1 - V]$, ou $P(T|\overline{A})$, représente la probabilité d'obtenir un faux positif parmi les échantillons d'intrus (personne classée comme autorisée alors qu'il s'agit d'un intrus). La justesse $Ju$ peut également être définie par $Ju = U.P(A) + V.[1 - P(A)]$ [257], où $P(A)$ représente la probabilité de l'occurrence d'une personne autorisée parmi toute la population d'échantillons testés. Dans les chapitres précédents, nous avons défini un nombre égal d'échantillons de tests positifs et négatifs. Par conséquent, $P(A) = 1/2$, et nous pouvons alors définir la justesse $Ju$ et la précision $Pr$ par l'équation suivante :

$$\begin{aligned} Pr &= \frac{U}{U - (1 - V)} \\ Ju &= \frac{U + V}{2} \end{aligned} \tag{6.1}$$

Soit $U_v$, $U_s$ et $U_e$ respectivement les sensibilités obtenues pour la modalité du visage, des vaisseaux sanguins et des empreintes digitales. Celles-ci représentent donc la probabilité d'obtention d'une classification correcte d'une personne autorisée, à partir de la modalité correspondante (données de référence), tel que $U_v = P(T_v|A_v)$, $U_s = P(T_s|A_s)$ et $U_e = P(T_e|A_e)$. De même, la probabilité de classer correctement un intrus est définie par $V_v = P(\overline{T_v}|\overline{A_v})$, $V_s = P(\overline{T_s}|\overline{A_s})$ et $V_e = P(\overline{T_e}|\overline{A_e})$, représentant respectivement la spécificité du visage, des vaisseaux sanguins et des empreintes. D'après Ludmila I. Kuncheva [257], il nous est alors possible d'estimer la sensibilité et la spécificité pour le vote à l'unanimité, $U_{una}$ et $V_{una}$, et pour le vote à la majorité, $U_{maj}$ et $V_{maj}$.

Pour valider l'authentification, le vote à l'unanimité nécessite l'acceptation simultanée des trois modalités. Les métriques peuvent donc être estimées par une simple multiplication des probabilités d'obtention d'une acceptation, pour chaque modalité. La sensibilité $U_{una}$ et la spécificité $V_{una}$ du vote à l'unanimité sont obtenues par les équations suivantes, dans le cas de l'évaluation générale où les tentatives d'intrusions $I$ sont représentées par des échantillons appartenant à un intrus (échantillons différents des données de référence).

$$\begin{aligned} U_{una} &= P(T_v|A_v) \times P(T_s|A_s) \times P(T_e|A_e) \\ U_{una} &= U_v \times U_s \times U_e \end{aligned} \tag{6.2}$$

$$\begin{aligned} V_{una} &= 1 - [1 - P(\overline{T_v}|\overline{A_v})] \times [1 - P(\overline{T_s}|\overline{A_s})] \times [1 - P(\overline{T_e}|\overline{A_e})] \\ V_{una} &= 1 - [1 - V_v] \times [1 - V_s] \times [1 - V_e] \end{aligned} \tag{6.3}$$

Toujours dans le cas où la classe non autorisée est représentée par des intrusions $I$, l'estimation des métriques d'évaluations du vote à la majorité sont plus complexes que pour le vote à l'unanimité. En effet, ce vote nécessite de prendre en compte les probabilités où au moins deux modalités sont détectées comme acceptées (pour le calcul de la sensibilité) et où au moins deux modalités sont détectées comme refusées (pour le calcul de la spécificité).

Le système d'équation 6.4 permet d'estimer la sensibilité $U^I_{maj}$ du vote à la majorité. Celle-ci représente la somme des probabilités d'acceptation simultanée du visage et des vaisseaux seuls, $P(T_v, T_s, \overline{T_e}|A_v, A_s, A_e)$, du visage et des empreintes seuls, $P(T_v, \overline{T_s}, T_e|A_v, A_s, A_e)$, des vaisseaux et des empreintes uniquement, $P(\overline{T_v}, T_s, T_e|A_v, A_s, A_e)$, ainsi que des trois modalités, $P(T_v, T_s, T_e|A_v, A_s, A_e)$. Le système d'équations 6.6 permet d'estimer la spécificité $V^I_{maj}$ du vote à la majorité. Celle-ci représente la somme des probabilités de refus des modalités, deux à deux, de la même manière que pour la sensibilité du vote à la majorité.

$$\begin{cases} X^U_0 &= P(T_v, T_s, T_e|A_v, A_s, A_e) &= U_v \times U_s \times U_e \\ X^U_1 &= P(T_v, T_s, \overline{T_e}|A_v, A_s, A_e) &= U_v \times U_s - P(T_v, T_s, T_e|A_v, A_s, A_e) \\ X^U_2 &= P(T_v, \overline{T_s}, T_e|A_v, A_s, A_e) &= U_v \times U_e - P(T_v, T_s, T_e|A_v, A_s, A_e) \\ X^U_3 &= P(\overline{T_v}, T_s, T_e|A_v, A_s, A_e) &= U_s \times U_e - P(T_v, T_s, T_e|A_v, A_s, A_e) \end{cases} \tag{6.4}$$

$$\begin{aligned} U^I_{maj} &= X^U_0 + X^U_1 + X^U_2 + X^U_3 \\ U^I_{maj} &= (U_v \times U_s) + (U_v \times U_e) + (U_s \times U_e) - 2.(U_v \times U_s \times U_e) \end{aligned} \tag{6.5}$$

$$\begin{cases} X^V_0 &= P(\overline{T_v}, \overline{T_s}, \overline{T_e}|\overline{A_v}, \overline{A_s}, \overline{A_e}) &= V_v \times V_s \times V_e \\ X^V_1 &= P(\overline{T_v}, \overline{T_s}, T_e|\overline{A_v}, \overline{A_s}, \overline{A_e}) &= V_v \times V_1 - P(\overline{T_v}, \overline{T_s}, \overline{T_e}|\overline{A_v}, \overline{A_s}, \overline{A_e}) \\ X^V_2 &= P(\overline{T_v}, T_s, \overline{T_e}|\overline{A_v}, \overline{A_s}, \overline{A_e}) &= V_v \times V_e - P(\overline{T_v}, \overline{T_s}, \overline{T_e}|\overline{A_v}, \overline{A_s}, \overline{A_e}) \\ X^V_3 &= P(T_v, \overline{T_s}, \overline{T_e}|\overline{A_v}, \overline{A_s}, \overline{A_e}) &= V_s \times V_e - P(\overline{T_v}, \overline{T_s}, \overline{T_e}|\overline{A_v}, \overline{A_s}, \overline{A_e}) \end{cases} \tag{6.6}$$

$$\begin{aligned} V^I_{maj} &= X^V_0 + X^V_1 + X^V_2 + X^V_3 \\ V^I_{maj} &= (V_v \times V_s) + (V_v \times V_e) + (V_s \times V_e) - 2.(V_v \times V_s \times V_e) \end{aligned} \tag{6.7}$$

Nous proposons également de traiter le cas d'une tentative d'intrusion particulière où l'intrus s'est procuré une "copie" d'une des trois modalités d'une personne autorisée. C'est à une photo du visage de la personne usurpée, ou une reproduction de son empreinte digitale, ou encore, bien que cela paraisse plus complexe, une reproduction de son doigt vascularisé. Il est alors possible de calculer un nouvel ensemble de métriques pour chaque modalité usurpée (la sensibilité restant cependant identique). Les cas $Usurp(v)$ / $Usurp(s)$ / $Usurp(e)$ décrivent respectivement une usurpation du visage, des vaisseaux ou des empreintes. En se basant sur les équations précédentes de la spécificité, il suffit alors de considérer la sensibilité de la modalité usurpée.

Les spécificités du vote à l'unanimité, pour l'usurpation de chacune des modalités, sont alors estimées par les équations suivantes :

$$\begin{aligned} V^{Usurp(v)}_{una} &= 1 - P(T_v|A_v) \times P(T_s|\overline{A_s}) \times P(T_e|\overline{A_e}) \\ V^{Usurp(v)}_{una} &= 1 - U_v \times [1 - V_s] \times [1 - V_e] \end{aligned} \tag{6.8}$$

$$\begin{aligned} V^{Usurp(s)}_{una} &= 1 - P(T_s|A_s) \times P(T_v|\overline{A_v}) \times P(T_e|\overline{A_e}) \\ V^{Usurp(s)}_{una} &= 1 - U_s \times [1 - V_v] \times [1 - V_e] \end{aligned} \tag{6.9}$$

$$\begin{aligned} V^{Usurp(e)}_{una} &= 1 - P(T_e|A_e) \times P(T_v|\overline{A_v}) \times P(T_s|\overline{A_s}) \\ V^{Usurp(e)}_{una} &= 1 - U_e \times [1 - V_v] \times [1 - V_s] \end{aligned} \tag{6.10}$$

De même, les spécificités du vote à la majorités, pour l'usurpation de chacune des modalités, sont alors estimées tel que le présentent les équations 6.11 où cette sensibilité est calculée dans le cas où le visage est usurpé.

$$\begin{cases} Z_0^V &= P(\overline{T_v}, \overline{T_s}, \overline{T_e}|A_v, \overline{A_s}, \overline{A_e}) &= [1-U_v] \times V_s \times V_e \\ Z_1^V &= P(\overline{T_v}, \overline{T_s}, T_e|A_v, \overline{A_s}, \overline{A_e}) &= [1-U_v] \times V_s - P(\overline{T_v}, \overline{T_s}, \overline{T_e}|A_v, \overline{A_s}, \overline{A_e}) \\ Z_2^V &= P(\overline{T_v}, T_s, \overline{T_e}|A_v, \overline{A_s}, \overline{A_e}) &= [1-U_v] \times V_e - P(\overline{T_v}, \overline{T_s}, \overline{T_e}|A_v, \overline{A_s}, \overline{A_e}) \\ Z_3^V &= P(T_v, \overline{T_s}, \overline{T_e}|A_v, \overline{A_s}, \overline{A_e}) &= V_a \times V_e - P(\overline{T_v}, \overline{T_s}, \overline{T_e}|A_v, \overline{A_s}, \overline{A_e}) \end{cases} \tag{6.11}$$

$$\begin{aligned} V_{maj}^{Usurp(v)} &= Z_0^V + Z_1^V + Z_2^V + Z_3^V \\ V_{maj}^{Usurp(v)} &= ([1-U_v] \times V_s) + ([1-U_v] \times V_e) + (Vs \times V_e) - 2.([1-U_v] \times V_s \times V_e) \end{aligned} \tag{6.12}$$

La tables 6.2 présente alors, à partir des résultats des chapitres précédents, les estimations des performances d'authentification par la fusion de décision appliquée aux votes à la majorité et à l'unanimité. La table 6.3 présente les estimations de ces mêmes métriques dans le cas où un intrus usurpe une des trois modalités de l'individu habilité de référence pour chacun des tests.

| **Type de fusion de décision** | **Se** ($\sim U$) | **Sp** ($\sim V$) | **Pr** | **Ju** |
|---|---|---|---|---|
| **Vote à la majorité** | 89,9 | 99,7 | 99,6 | 94,8 |
| **Vote à l'unanimité** | 51,7 | 100 | 100 | 75,8 |

Table 6.2 – Estimation des performances d'authentification, en pourcentage, obtenues par fusion de décision des trois modalités à partir des résultats des chapitres précédents

| **Type de fusion de décision** | **Modalité usurpée** | **Se** ($\sim U$) | **Sp** ($\sim V$) | **Pr** | **Ju** |
|---|---|---|---|---|---|
| **Vote à la majorité** | **Visage** | 89,9 | 96,9 | 96,7 | 96,1 |
| | **Vaisseaux** | 89,9 | 91,5 | 91,3 | 93,3 |
| | **Empreintes** | 89,9 | 92,3 | 92,1 | 93,8 |
| **Vote à l'unanimité** | **Visage** | 51,7 | 100 | 100 | 75,8 |
| | **Vaisseaux** | 51,7 | 99,8 | 99,7 | 75,8 |
| | **Empreintes** | 51,7 | 99,9 | 99,9 | 75,8 |

Table 6.3 – Estimation des performances d'authentification, en pourcentage, obtenues par fusion de décision des trois modalités avec usurpation d'une modalité à la fois, à partir des résultats des chapitres précédents

Les performances générales des modalités indépendantes sont rappelées par la table 6.1. La fusion de décision par vote à la majorité permet d'atteindre 90% de sensibilité avec une spécificité proche 100%. En d'autres termes, très peu d'intrus peuvent déverrouiller la porte et 10% des personnes habilitées subissent un refus d'authentification. Le vote à l'unanimité en revanche présente un taux de faux refus proche de 50% pour aucune fausse acceptation.

Dans le cas où un intrus a été en mesure de dupliquer une des modalités d'une personne habilitée, le vote à la majorité ne permet pas de conserver un taux d'intrusion nul. En effet, dans le cas d'une usurpation du visage d'une personne autorisée (masque, photo, etc), la spécificité perd 3%. Avec une copie des vaisseaux sanguins ou des empreintes digitales, la spécificité du vote à la majorité diminue alors jusqu'à environ 92%. En revanche, le vote à l'unanimité permet conserver un taux d'intrusion quasiment nul dans le cas où une modalité est copiée.

### Validation de la fusion de modalités

Afin de réaliser cette étude expérimentale, nous avons construit une nouvelle base de données à partir des ensembles d'images de chaque modalité. À chaque personne sont associées 5 images de son visage avec son classifieur personnel comme référence, et 5 ensembles de descripteurs (lignes et points singuliers) ainsi qu'un ensemble de référence pour les empreintes et les réseaux vasculaires. Lors de l'authentification d'un individu, un ensemble de test, formé des trois modalités, est comparé aux références associées à cette personne. Trois prédictions sont alors obtenues, une pour chaque modalité, définissant si le sujet de test est classé comme autorisé ou intrus. La décision globale est alors basée sur une fusion de décision, fournie de deux manières différentes.
Cette nouvelle base de données a été contrainte à un nombre restreint de personnes et d'échantillons relativement aux possibilités offertes par la fusion des bases de données de nos trois modalités (11 personnes disponibles pour les visages et 6 ensembles de descripteurs par personne pour les vaisseaux sanguins). Les éléments composant cette nouvelle base de données ont été sélectionnés aléatoirement parmi les divers éléments disponibles. Cette base de données est formée de 11 individus, et chacun possède les trois modalités pour lesquelles on distingue un échantillon de référence et 5 autres dédiés aux tests. Par conséquent, les nouveaux tests effectués ne représentent pas l'ensemble des possibilités exploitées précédemment par les modalités individuelles. Les résultats présentés proviennent donc d'une moyenne de 1375 tentatives d'authentification positives et 1375 tentatives d'intrusions. Comme le montre la table 6.4, les performances individuelles de chaque modalité, sur nos nouvelles données, diffèrent des résultats présentés jusque là : les performances d'authentification du visage sont réduites, et celles des empreintes digitales sont maximales.

| **Modalité** | **Se** ($\sim U$) | **Sp** ($\sim V$) | **Pr** | **Ju** |
|---|---|---|---|---|
| **Visage** | 62,0 | 84,0 | 79,5 | 73,0 |
| **Empreintes** | 100 | 100 | 100 | 100 |
| **Vaisseaux** | 86,0 | 95,2 | 94,7 | 90,6 |

TABLE 6.4 – Performances d'authentification obtenues pour chaque modalité avec le nouvel ensemble réduit d'évaluation

La table 6.5 présente les métriques d'évaluations décrivant les performances d'authentification de la fusion de décision des trois modalités biométriques pour le vote à la majorité et le vote à l'unanimité. La table 6.6 présente la robustesse aux tentatives d'usurpation d'une modalité parmi les trois. Ces deux tables présentent donc les performances d'authentification par fusion de décision à partir des nouvelles données réduites.
La fusion de décision par vote à la majorité permet d'atteindre 95% de sensibilité avec une spécificité de 100%. En d'autres termes, aucun intrus ne peux déverrouiller la porte et seul 5% des personnes habilitées subissent un refus d'authentification. Le vote à l'unanimité en revanche présente un taux de faux refus proche de 50% pour un taux nul de fausse acceptation.

| Type de fusion de décision | Se ($\sim U$) | Sp ($\sim V$) | Pr | Ju |
|---|---|---|---|---|
| **Vote à la majorité** | 95,2 | 100 | 100 | 97,6 |
| **Vote à l'unanimité** | 52,8 | 100 | 100 | 76,4 |

TABLE 6.5 – Performances d'authentification, en pourcentage, obtenues par fusion de décision des trois modalités sur notre nouvel ensemble de données réduit

| Type de fusion de décision | Modalité usurpée | Se ($\sim U$) | Sp ($\sim V$) | Pr | Ju |
|---|---|---|---|---|---|
| **Vote à la majorité** | **Visage** | 95,2 | 96,9 | 96,8 | 96,1 |
| | **Vaisseaux** | 95,2 | 85,1 | 86,5 | 90,2 |
| | **Empreintes** | 95,2 | 79,6 | 82,4 | 87,4 |
| **Vote à l'unanimité** | **Visage** | 52,8 | 100 | 100 | 76,4 |
| | **Vaisseaux** | 52,8 | 100 | 100 | 76,4 |
| | **Empreintes** | 52,8 | 100 | 100 | 76,4 |

TABLE 6.6 – Performances d'authentification, en pourcentage, obtenues par fusion de décision des trois modalités avec usurpation de l'une d'entre elles, sur notre nouvel ensemble de données réduit

Dans le cas où un intrus a été en mesure de dupliquer une des modalités d'un individu habilité, le vote à la majorité ne permet pas de conserver un taux d'intrusion nul. En effet, pour l'usurpation du visage d'une personne autorisée (masque, photo, etc), la spécificité perd 3%, et 5% si ce sont les vaisseaux sanguins qui sont usurpés. Du fait de la faible spécificité du visage, si les empreintes sont copiées, alors la spécificité du vote à la majorité chute à 80%. En revanche, le vote à l'unanimité permet conserver un taux d'intrusion nul dans le cas où une modalité est copiée.

### 6.1.3/ DISCUSSION

Bien que la sensibilité estimée à partir des résultats des chapitres précédents soit légèrement plus faible que la sensibilité obtenue expérimentalement sur un sous ensemble de données, les performances d'authentification, reportées dans les tables 6.5, 6.6, 6.2 et 6.3, sont similaires. Par conséquent, même sur un jeu de données réduit, les valeurs présentées dans ces tableaux sont stables. Le vote à la majorité représente un bon compromis entre fausse acceptation et faux rejet. Le vote à l'unanimité peut être une option envisageable dans le cas où le taux d'intrusion est un paramètre critique dont le risque doit être nul. De plus, en considérant un vote à la majorité sur un plus grand nombre $L$ de classifieur (divers classifieurs par modalités ou plus de modalités étudiées), la restrictivité du système peut être ajustable. Tel qu'illustré par la figure 6.2, la définition formelle du vote d'une population est présentée par l'équation 6.13 [257], où $\omega_k$ représente la classe issue du vote, et $d_{i,k}$ représente la prédiction du classifieur $i$. Cette équation montre que la quantité $\alpha.L$, définissant la proportion de votes pour une classe, peut être paramétrée par l'opérateur $\epsilon$. Si $\alpha = 1$, cela correspond au vote à l'unanimité et si $\alpha = (\frac{L}{2}+1)$, cela correspond à la majorité simple. Sinon, cela représente la majorité qualifiée.

$$\begin{cases} \omega_{k=1}, & \text{Si } \sum_{i=1}^{L} d_{(i,k=1)} \geq \alpha.L \\ \omega_{k=0}, & \text{Sinon} \end{cases} \quad \text{où : } \begin{cases} \alpha = & \frac{1}{2}+\epsilon \\ \alpha \in & ]0;1] \end{cases} \tag{6.13}$$

## 6.2/ Implantation matérielle

### 6.2.1/ Smart Caméra - Intelligence embarquée

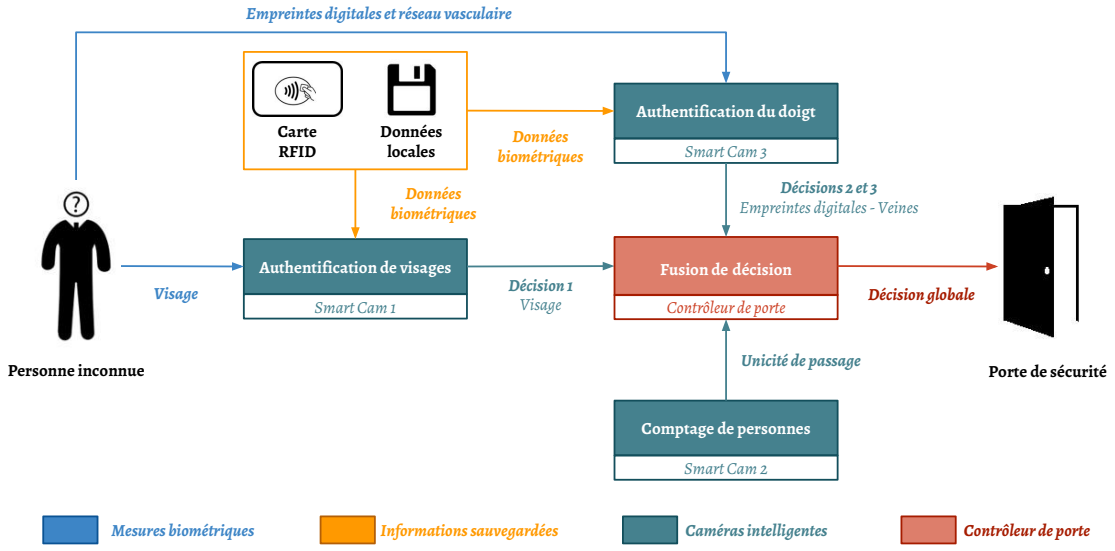



Figure 6.3 – Schéma bloc du système complet

Le système de sécurité biométrique complet comprend les différents blocs de traitement, figure 6.3, assurés par des caméras intelligentes. Chacun des blocs en vert de la figure 6.3 représente une "smart caméra" permettant d'effectuer à la fois l'acquisition et les différents traitements pour chaque attribut biométrique. Ainsi, seuls les résultats de ces traitements transitent sur le réseau, les données personnelles (comme les images du sujet) étant strictement limitées à la caméra. Quelle que soit son architecture ou son objectif, une caméra est définie comme intelligente dès lors qu'elle embarque une partie calculatoire afin de traiter directement l'image acquise.

Les images acquises par le capteur sont donc traitées au sein même de la caméra pour en extraire l'information pertinente. Les résultats sont transmis au système global via les interfaces de communication intégrées. La cible matérielle en charge des traitements embarqués peut être de nature variée. Il peut s'agir de circuits intégrés dédiés à une tâche non modifiable ou de cibles plus flexibles permettant de modifier la nature des traitements appliqués. Il existe deux grandes familles de cibles matérielles permettant de modifier les traitements : les circuits séquentiels programmables (processeurs, GPU, DSP...) ou les circuits logiques reconfigurables (FPGA). Ces derniers sont particulièrement efficaces pour l'implantation de tâches régulières et possèdent un fort potentiel de parallélisme de données que nous avons exploité dans des travaux antérieurs [258, 259, 260]. La puissance de calcul de ce type d'unités est donc très importante, cependant les temps de développement sont conséquents. Il est envisageable de les réduire par l'utilisation de méthodes de prototypage rapide comme nous avons pu le mettre en pratique pour les cibles FPGA [261, 262] ou avec des cibles hybrides réunissant des ressources logiques reconfigurables aux processeurs et interfaces de communications tels que les Systèmes sur Puces (*Systems on Chip SoC*) [263]. Cependant nous avons privilégié le choix d'une cible séquentielle programmable de type processeur multi-coeurs. Le compromis entre performance de calcul et flexibilité nous a paru plus intéressant vis à vis de notre application, afin d'accroître la souplesse du système. De plus, le faible coût de ce type de cibles matérielles permet de multiplier leur nombre à moindre frais, conformément à nos besoins, voir même de réaliser de véritables réseaux de caméras intelligentes [264].

Ici, les caméras intelligentes 1 et 2 sont composées d'une Raspberry PI3 (unité de calcul et stockage interne) et d'un capteur d'image standard. Très abordable financièrement, cette carte présente des capacités de calculs plus limitées qu'une machine de développement (table 6.7). Elle demeure néanmoins capable de remplir les fonctions qui lui ont été attribuées.

| | |
|---|---|
| Processeur de calculs | ARM Cortex-A53 64 bits ; 4 coeurs ; 1,2 GHz |
| Mémoire | 1 Go ; LPDDR2 |
| Stockage | MicroSD |
| Ports USB 2.0 | 4 |
| Réseau | 10/100 Ethernet ; Wifi 802.11n ; Bluetooth 4.1 BLE |
| Dimensions | 8,56 cm × 5,40 cm × 1,70 cm |
| Alimentation | 5V / 2,5A ; micro USB |
| Prix | 35€ à 40€ |

TABLE 6.7 – Caractéristiques matérielles de la carte Raspberry PI 3 modèle B

La carte Raspberry PI3 nous paraît donc présenter un bon compromis entre puissance de calcul, coûts de déploiement, encombrement et modularité. Son stockage interne est assuré par une carte micro SD de capacité 32Go, sur laquelle est installé un système d'exploitation Raspbian, basé sur une distribution Debian (noyau linux) optimisée pour fonctionner sur une carte Raspberry. De manière générale, nos caméras intelligentes contiennent les éléments nécessaires aux calculs de nos méthodes, comme leurs scripts et programmes. La caméra intelligente 1, dédiée à l'authentification de visages, contient également les espaces de visages issus de l'ACP ainsi que les projections pré-calculées de la classe des intrus. Les acquisitions des images sont assurées soit par une webcam traditionnelle connectée à la carte en USB (configuration utilisée pour l'étude précédente), soit par une caméra dédiée à la carte Raspberry branchée via une nappe sur le port CSI.


(a) Raspberry et Capteur (b) Boitier vue de face (c) Boitier vue de coté

FIGURE 6.4 – Boitier de la Smart Caméra Raspberry PI3 avec le capteur d'images CSI Raspberry

La carte Raspberry ne comporte par défaut aucun dissipateur thermique. Hors, nos premières expérimentations ont montré que l'utilisation d'un CNN entraîne une charge trop importante pour le processeur et donc une montée en température de la carte. Bien que les performances soient réduites en contrôlant la puissance maximale du processeur afin de réduire la température, les longs calculs d'apprentissage (cf section suivante) entraînent une perte de réponse du système qui cesse alors de fonctionner. Afin de pallier a ce problème et faciliter le déploiement des caméras intelligentes en conditions réelles, nous utilisons un boitier dédié aux cartes Raspberry, conçu au sein de

notre laboratoire (figure 6.4). Ce boitier permet l'accès aux différentes entrées et sorties, et intègre un ventilateur alimenté par la carte. Celui-ci permet alors de limiter la montée en température et de garantir le bon fonctionnement du système avec des performances constantes. De plus, la structure externe du boitier présente une connexion mécanique de type liaison rotule offrant la possibilité de régler facilement les orientations de la caméra. L'intégration de la caméra au boitier est simplifiée par l'utilisation de la caméra Raspberry CSI. Celle ci présente des propriétés équivalentes à celles de la webcam utilisée pour l'étude précédente qui reste donc valable.

Les plateformes embarquées telles que la Raspberry atteignent rapidement leurs limites dès lors qu'elles sont réquisitionnées pour des CNN. Cependant, de nouvelles plateformes de calculs additionnelles ont vu le jour très récemment, permettant, à des coûts réduits (≤100€), de déporter ces calculs. L'une d'elles, nommée "Movidius [265]" (société Intel), se présente sous la forme d'une clé USB à connecter sur la carte Raspberry. Composée d'un VPU *(Vision Processor Unit* [266, 267]), microprocesseur optimisé pour la vision, la version "Myriad 2" [268] ne consomme qu'1W et promet une puissance de calculs de 100 GFlops [269]. En comparaison, ramenée à une consommation d'1W, la puissance de la Raspberry PI3 est de 0,8 GFlop [270]. Ces avancées technologiques permettent de dépasser les limitations actuelles pour des applications dédiées [271].

Les traitements d'images des expérimentations suivantes sont réalisés via la librairie OpenCV [272, 273]. Celle-ci fourni les fonctions permettant d'appliquer l'algorithme de Viola-Jones [135] pour détecter les visages, d'effectuer les différents filtrages de l'image, d'extraire les zones d'intérêt et de les redimensionner. Les méthodes classiques de Machine Learning (ACP, SVM, RF) sont également fournies par cette librairie. De très nombreux environnements de développement (*framework*) sont disponibles pour exécuter les méthodes de Deep Learning/Transfer Learning. Parmi les plus connus, on retrouve Caffe/Caffe2 [274], Chainer [275], CNTK [276], Theano [277], Tensorflow [144], etc. Au regard de la popularité de ces framework sur la plateforme github [278], tableau 6.8, nous avons choisi d'utiliser Tensorflow. En effet, au delà de sa simplicité d'installation et d'utilisation, il est largement utilisé par la communauté scientifique, ce qui permet d'obtenir plus facilement des informations quant aux potentiels problèmes liés aux évolutions du framework. Malgré tout, il ne figure pas parmi les plus rapides [279, 280], surtout en considérant un unique cœur de calcul sur CPU. C'est pourquoi, les temps de calculs de l'implantation du Transfer Learning dans la section suivante ne dépendent pas seulement de la plateforme cible, mais également du framework utilisé.

| **Frameworks** | **Favoris** ***(Star)*** | **Observateurs** ***(Watcher)*** | **Copie** ***(Fork)*** | **Date de création** |
|---|---|---|---|---|
| caffe2 | 8300 | 577 | 2105 | 25/06/2015 |
| chainer | 4141 | 325 | 1099 | 05/06/2015 |
| CNTK | 15150 | 1372 | 4046 | 26/11/2015 |
| tensorflow | 110580 | 8400 | 67957 | 07/11/2015 |

TABLE 6.8 – Popularité sur github, suivant trois type d'informations statistiques, de framework Deep Learning [278] - Mise à jour Sept. 2018

La forte connotation industrielle du projet sur lequel repose cette thèse impose d'évaluer les caractéristiques du système au travers une implémentation réelle plutôt que par une estimation théorique de la complexité algorithmique des méthodes mises en œuvre. Les algorithmes présentés dans les chapitres précédents sont donc intégrés sur la carte Raspberry PI 3 décrite précédemment. Dans la suite de cette section, nous présentons alors les temps de calculs des différents traitements constituant les protocoles d'authentification du visage et des modalités du doigt.

### 6.2.2/ Authentification de visages - Implantation sur caméra intelligente

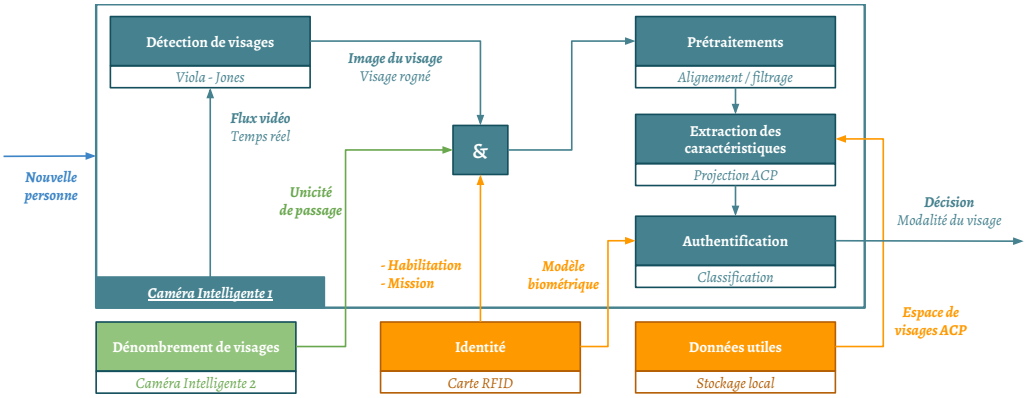



Figure 6.5 – Protocole complet de l'authentification de visages du système industriel

Dans le chapitre 4, nous avons décrit et confronté différentes méthodes de classification, permettant d'effectuer une authentification de visages. Seules les performances brutes ont été présentées, évaluées par les métriques de sensibilité, spécificité, précision et justesse, indépendamment de la plateforme de calcul. L'objectif ici est de vérifier les contraintes de temps, considérant une implantation sur une architecture embarquée. Pour chaque phase des méthodes décrites dans le chapitre 4, cette section expose donc les temps de calculs moyens de leur implantation embarquée sur la caméra intelligente décrite précédemment. Quelle que soit la valeur du paramètre $\beta$ (valeur non nulle), l'image est d'abord filtrée puis sommée à l'image originale. Ce filtrage est d'environ 3 ms pour le filtre de Sobel et 8 ms pour le filtre de TanTriggs. La détection de visage, permise par l'algorithme de Viola-Jones, est effectuée image par image à partir du flux vidéo de 640x480 pixels fourni par la caméra intelligente. Sous échantillonner l'image en niveau de gris, par un facteur trois, permet d'effectuer cette détection en 40 ms.

L'observation des différentes étapes de traitements des méthodes de Machine Learning traditionnel montre que les temps de calculs ne dépendent pas uniquement de la vitesse des classifieurs. On peut citer en particulier le chargement de l'espace de visages et la projection de nouvelles images dans cet espace. Les temps de projection moyens d'une image ($100 \times 100$ pixels) dans l'espace de visages sont donnés par le tableau 6.9 (colonne Proj), ainsi que le temps de chargement de cet espace (colonne Ch. ACP). De la même manière que précédemment, les extrémums de ces valeurs apparaissent en couleur. Les temps de calculs les plus longs (les moins bons), sont colorés en rouge pour chaque type de filtrage, et sont mis en gras pour la valeur extrême parmi toutes les configurations de filtrage pour un seuil d'ACP donné. La représentation des meilleurs temps de calculs (en bleu) respecte les mêmes règles. En conditions réelles, pour une configuration d'ACP donnée, cet espace de visage est chargé uniquement au démarrage du système. Le temps de chargement, jusqu'à plusieurs secondes selon les configurations, est donc présenté à titre informatif et n'est pas à prendre en compte dans les calculs des temps d'apprentissage et d'authentification.

On remarque, en corrélant le tableau 6.9 avec le tableau 4.2, que ces temps de projection dépendent effectivement de la configuration d'entrée (contribution de filtrage et seuil de l'ACP), mais seul le nombre de valeurs propres conservées influe sur ces temps de calculs (nombre de vecteurs propres constituant l'espace de visage). Cela confirme le fait que le temps de projection d'une image dans l'espace de visage est directement lié au nombre de valeurs propres conservées dans cet espace.

| Filtrage | PCA 90% - [ms] | | PCA 70% - [ms] | | PCA 50% - [ms] | |
|---|---|---|---|---|---|---|
| | Ch. ACP | Proj. | Ch. ACP | Proj. | Ch. ACP | Proj. |
| Pas de filtrage | **2860** | **53** | **596** | **11** | **206** | **2** |
| Sobel, $\beta$ = 20 | 3281 | 60 | 672 | 14 | 211 | 5 |
| Sobel, $\beta$ = 40 | 4481 | 81 | 1179 | 25 | 313 | 7 |
| Sobel, $\beta$ = 60 | 5691 | 113 | 2193 | 39 | 771 | 18 |
| Sobel, $\beta$ = 80 | 6184 | 123 | 2779 | 48 | 1174 | 24 |
| Sobel, $\beta$ = 100 | 6189 | 125 | 2818 | 50 | 1215 | 25 |
| TanTriggs, $\beta$ = 20 | 4253 | 92 | 1042 | 26 | 287 | 10 |
| TanTriggs, $\beta$ = 40 | 5727 | 114 | 2052 | 41 | 588 | 19 |
| TanTriggs, $\beta$ = 60 | 6741 | 138 | 3307 | 59 | 1421 | 36 |
| TanTriggs, $\beta$ = 80 | 7341 | 152 | 4196 | 81 | 2228 | 51 |
| TanTriggs, $\beta$ = 100 | **7575** | **157** | **4524** | **87** | **2582** | **57** |

TABLE 6.9 – Temps de chargement des espaces de visages et temps de calcul des projections d'une image selon la contribution du filtrage et le seuil d'ACP

Les temps de projection sont compris entre 2 ms pour 7 valeurs propres (pas de filtrage et ACP 50%) et 157 ms pour 299 valeurs propres (TanTriggs 100% et ACP 90%). Afin de déterminer les temps de calculs de manière réaliste, il est nécessaire de prendre en compte le temps de filtrage de l'image avant la projection. Le traitement de plusieurs échantillons consécutifs peut en effet représenter un temps de calcul non négligeable.

#### 6.2.2.1/ TEMPS DE CALCULS DE LA PHASE D'ENRÔLEMENT

Lors de la phase d'enrôlement, les méthodes de Machine Learning classique commencent par charger l'espace de visages issu de l'ACP. Pour un certain espace de visage chargé, les projections de 40 personnes non autorisées ou intrus (calculées au préalable) sont ensuite mises en mémoire. S'en suit l'acquisition de 40 images du visage du sujet et l'application du filtrage correspondant à l'espace de visages qui été chargé, pour finalement les y projeter. Les vecteurs de projections résultants (40 vecteurs de $\lambda$ éléments) sont ensuite fusionnés avec ceux des usurpateurs (40 vecteurs de $\lambda$ éléments) pour former la matrice des données d'apprentissage de dimensions $80 \times \lambda$. Un vecteur d'étiquettes définissant la classe de chacune de ces 80 projections est également créé. Le couple formé par cette matrice et ce vecteur est alors utilisé par le classifieur pour l'apprentissage automatique.
Par conséquent, outre les temps de calculs de l'apprentissage des classifieurs, présentés dans le tableau 6.11, le calcul de la durée moyenne d'un enrôlement complet doit prendre en considération les temps d'un certain nombre d'opérations non négligeables. Ces opérations, présentées dans le tableau 6.10, couvrent le filtrage des images acquises ainsi que leur projection (colonne F+P) et le chargement des projections des personnes non autorisées (colonne Ch. PI).

| Filtrage | PCA 90% - [ms] | | PCA 70% - [ms] | | PCA 50% - [ms] | |
|---|---|---|---|---|---|---|
| | 40×(F+P) | Ch. PI | 40×(F+P) | Ch. PI | 40×(F+P) | Ch. PI |
| Pas de filtrage | **2221** | **16** | **436** | **4** | **69** | 2 |
| Sobel, $\beta$ = 20 | 2531 | 18 | 688 | 5 | 316 | **2** |
| Sobel, $\beta$ = 40 | 3360 | 24 | 1105 | 7 | 388 | 3 |
| Sobel, $\beta$ = 60 | 4629 | 30 | 1698 | 13 | 835 | 5 |
| Sobel, $\beta$ = 80 | 5057 | 32 | 2062 | 15 | 1064 | 7 |
| Sobel, $\beta$ = 100 | 5100 | 32 | 2114 | 16 | 1108 | 7 |
| TanTriggs, $\beta$ = 20 | 4896 | 23 | 1340 | 7 | 734 | 2 |
| TanTriggs, $\beta$ = 40 | 5842 | 30 | 1942 | 12 | 1066 | 4 |
| TanTriggs, $\beta$ = 60 | 6383 | 35 | 2691 | 18 | 1746 | 9 |
| TanTriggs, $\beta$ = 80 | **6586** | 38 | 3541 | 22 | 2353 | 13 |
| TanTriggs, $\beta$ = 100 | 6582 | **39** | **3809** | **28** | **2617** | **15** |

TABLE 6.10 – Temps de préparation des données pour l'apprentissage : Filtrage et projection de 40 images (F+P) + chargement des projections d'intrus (PI)

D'après la table 6.10 on remarque que le temps de chargement des projections de la seconde classe est relativement court avec moins de 40 ms. Suivant la configuration, le filtrage et la projection des 40 images acquises peuvent atteindre au maximum 6,5 s, et 70 ms au minimum. Cette table ne comprend pas les temps d'acquisition des images, que nous définissons comme la phase de détection et d'extraction du visage à partir des images du flux vidéo de la caméra (représentant 40 ms). Le temps d'acquisition de 40 images de visage dépend d'une temporisation ($T$) que l'on souhaite appliquer entre deux clichés. En effet, afin d'augmenter la variance interclasse dans les données d'apprentissage, il n'est pas judicieux d'acquérir 40 images consécutives. Par conséquent, on peut considérer un temps d'acquisition environ égal à $40 \times (T + 40)$ ms ($= 40 \times T + 1600$ ms).

L'estimation du temps total de la préparation des données pour l'apprentissage est complexe. Comme nous l'avons vu, cette préparation consiste en différentes tâches, certaines étant regroupées en processus séquentiel (comme le filtrage et la projection de l'image). Cependant, il n'est pas nécessaire d'attendre d'avoir acquis toutes les images du visage pour procéder à leur filtrage et à leur projection dans l'espace de visage associé. Ces deux dernières tâches peuvent être effectuées sur une image, en parallèle de l'acquisition des suivantes.

L'observation des temps de calculs présentés table 6.11 montre que pour les SVM, l'apprentissage est systématiquement plus long que pour les RF. Ces temps de calculs augmentent, pour ces deux classifieurs, à mesure que le nombre de valeurs propres augmente (ce qui augmente par conséquent la dimension de la matrice d'apprentissage). L'apprentissage des SVM est compris entre 314 ms et 2573 ms, tandis que l'apprentissage des RF est compris entre 15 ms et 956 ms. En comparaison, les temps de calculs dédiés à la préparation des données ( "40x(F+P)"+"Ch. PNA" ) sont compris entre 71 ms (aucun filtre et ACP à 50%) et 6624 ms (TanTriggs, $\beta$ = 80 et ACP à 90%). Cela représente un temps 3 à 7 fois supérieur à la durée nécessaire à la phase d'apprentissage des RF. La durée de l'apprentissage des SVM est généralement inférieure au temps de préparation des données, atteignant jusqu'à un facteur deux.

| Filtrage | PCA 90% - [ms] | | PCA 70% - [ms] | | PCA 50% - [ms] | |
|---|---|---|---|---|---|---|
| | SVM | RF | SVM | RF | SVM | RF |
| Pas de filtrage | **1029** | **199** | **400** | 52 | 322 | 25 |
| Sobel, $\beta$ = 20 | 1141 | 229 | 409 | 40 | **314** | **15** |
| Sobel, $\beta$ = 40 | 1474 | 287 | 548 | 94 | 329 | 29 |
| Sobel, $\beta$ = 60 | 1807 | 569 | 823 | 144 | 430 | 69 |
| Sobel, $\beta$ = 80 | 1954 | 517 | 985 | 179 | 540 | 136 |
| Sobel, $\beta$ = 100 | 1964 | 476 | 1013 | 205 | 550 | 107 |
| TanTriggs, $\beta$ = 20 | 1427 | 218 | 504 | **37** | 321 | 23 |
| TanTriggs, $\beta$ = 40 | 1817 | 377 | 782 | 137 | 392 | 56 |
| TanTriggs, $\beta$ = 60 | 2144 | 404 | 1166 | 193 | 629 | 95 |
| TanTriggs, $\beta$ = 80 | 2409 | 619 | 1466 | 377 | 871 | 159 |
| TanTriggs, $\beta$ = 100 | **2573** | **956** | **1633** | **493** | **1015** | **279** |

TABLE 6.11 – Temps d'apprentissage des SVM et RF selon les contributions de filtrage et les seuils d'ACP

Afin de simplifier les calculs nous considérons les images du sujet comme étant déjà présentes dans la mémoire de la caméra intelligente. En prenant les extrémums de ces temps de calculs, le temps moyen d'un enrôlement complet est compris entre 393 ms et 9193 ms pour les SVM et entre 97 ms et 7576 ms pour les RF. En considérant la détection et l'extraction des quarante images de visage du sujet par la caméra intelligente, dans le cas le plus simple (c'est à dire sans temporisation et sans parallélisation), ces extrémums atteignent 1993 ms et 10793 ms pour les SVM, ainsi que 1697 ms et 9176 ms pour les RF. Dans le pire des cas, l'enrôlement n'excède pas 11 s.

Une fois l'apprentissage effectué, les paramètres du classifieur doivent être réduits et le modèle résultant doit être compressé pour être ensuite stocké sur la carte sans contact. Le temps nécessaire à la sauvegarde locale des modèles biométriques varie entre 2 ms et 76 ms pour les SVM et entre 11 ms et 43 ms pour les RF. La réduction de la précision des paramètres n'excède pas 572 ms pour le modèle le plus lourd des SVM, et 171 ms pour le modèle le lourd des RF. L'algorithme de compression sélectionné, BZip2, est également une étape à prendre en compte au delà de la simple création du modèle biométrique. Cette compression n'excède pas 150 ms pour le modèle le lourd des SVM (après RPP), et 1100 ms pour le modèle le lourd des RF (après RPP).

La dernière étape consiste à écrire les données de ce modèle biométrique dans la carte sans contact. Les cartes RFID les plus répandues, de type MIFARE DESFire Hautes Fréquences [1] (HF), disposent d'une mémoire utilisateur jusqu'à 8Ko pour une vitesse de transfert de 3,12 Ko/s à une fréquence de fonctionnement de 13,56 MHz. Celles disposant de la plus grande quantité de mémoire [2], atteignant 64 Ko, sont catégorisées Ultra Hautes Fréquences (UHF). Leur vitesse de transfert est de 28 Ko/s pour un fonctionnent à 860-960MHz. Cependant, ces cartes sont bien plus imposantes (dimensions et poids) et sont dédiées à des domaines d'application particuliers.

1. Site web : sbedirect - "Comprendre la RFID en 10 points"
2. Documentation technique Fujitsu - *"World's Largest Capacity 64KByte FRAM Metal Mount RFID Tag"*

Nous n'avons pas eu l'occasion d'implémenter notre système complet en utilisant les cartes sans contact. Il est néanmoins possible d'estimer le temps d'écriture. En considérant donc une implantation MIFARE DESFire, et le modèle biométrique le plus lourd parmi ceux sélectionnés dans la table 4.6, c'est à dire 7,5 Ko, la lecture et l'écriture devrait s'effectuer en 2,2 s. Ces processus atteignent 540 ms pour le modèle biométrique sélectionné le plus léger (1,9 Ko).

Lors de l'apprentissage d'un CNN, les images propagées dans le réseau sont sélectionnées aléatoirement par groupe (*batch*) afin d'effectuer l'ajustement des paramètres par la descente de gradient. Par conséquent, de nombreuses images peuvent être utilisées plusieurs fois. En ce qui concerne le Transfer Learning, seules les couches de classification sont ajustées. Les couches précédentes étant gelées, l'extraction de caractéristiques produit les mêmes données à chaque passage de la même image. Dans une démarche d'optimisation, l'environnement de développement utilisé intègre une étape préliminaire (une dernière couche nommée "*bottleneck*"). Elle permet, en propageant une première fois toutes les images d'apprentissage au travers des couches gelées, de mémoriser les caractéristiques détectées pour chacune de ces images et de fournir un "résumé" compact des caractéristiques extraites. Par la suite, ce sont ces caractéristiques qui sont utilisées pour alimenter les dernières couches, évitant alors de les calculer plusieurs fois inutilement et économisant ainsi les ressources mémoire et temporelles. La table 6.12 présente les temps d'apprentissage des CNN étudiés comprenant cette étape intermédiaire, ainsi que le temps de quantification du réseau (de type *"weight rounded"*), après son ré-apprentissage.

| **Réseau** | **1000 images par classe - [s]** | | **40 images par classe - [s]** | |
|---|---|---|---|---|
| | Apprentissage | Quantification | Apprentissage | Quantification |
| v1_0.25_128 | **3680** | **11** | 2947 | **11** |
| v1_0.25_224 | 3775 | 11 | 3034 | 11 |
| v2_0.35_128 | 3860 | 13 | **2942** | 13 |
| v2_0.35_224 | **3978** | **13** | **2970** | **13** |

TABLE 6.12 – Durées de l'apprentissage des réseaux MobileNet v1 et v2 les plus légers, respectivement pour 1000 et 40 images d'apprentissage par classe, de résolution 128x128 et 224x224.

Avec un millier d'images par classe, le processus d'apprentissage est considérablement plus long que le Machine Learning traditionnel, dépassant légèrement l'heure de calcul que ce soit pour la version 1 ou 2 des réseaux MobileNet. Pour l'apprentissage, la version 2 est plus lente de quelques minutes que la version 1. Les phases post-apprentissage ne nécessitent qu'une dizaine de secondes. L'utilisation de 40 images par classe permet de réduire d'une dizaine de minutes les temps d'apprentissage, et les deux versions des réseaux sont désormais équivalentes en temps de calcul, n'excédant pas les 3000 secondes environ (50 minutes). L'architecture des réseaux n'étant pas modifiée par le ré-apprentissage du Transfer Learning, il est normal de retrouver la même durée pour la phase de quantification post-apprentissage pour ces deux configurations du nombre d'images d'entrée.

Comme précisé précédemment lors de la présentation de la caméra intelligente, cette implémentation des CNN sur Raspberry entraîne des dysfonctionnements. En effet, la charge processeur est très importante et entraine une montée en température, qui, du fait de la durée de l'apprentissage de l'ordre de l'heure de calcul, cause la perte de réponse du système. Au delà d'un point de vue purement pratique en implantation industrielle, l'utilisation du boitier proposé intégrant un ventilateur est donc indispensable, même en phase d'expérimentation.

#### 6.2.2.2/ Temps de calculs de la phase authentification

Les tables 6.14 et 6.15 présentent cette fois ci les temps de calcul de phase décisionnelle. Les résultats présentés sont les moyennes de temps obtenus pour les 11 sujets au travers de 2000 prédictions (1000 échantillons testés par classe). De la même manière que précédemment, pour considérer un fonctionnement temps réel, la durée de l'authentification complète ne doit pas se limiter aux seuls temps de calculs des prédictions. Pour le Machine Learning classique, le chargement du modèle biométrique, la détection et l'extraction du visage, puis son filtrage optionnel ainsi que sa projection dans la base de visages (tableau 6.13) doivent être considérés. Pour les CNN, seuls le chargement du réseau et la détection de visage peuvent être pris en compte.

| **Filtrage** | **PCA 90% - [ms]** | **PCA 70% - [ms]** | **PCA 50% - [ms]** |
|---|---|---|---|
| Pas de filtrage | **53** | **11** | **2** |
| Sobel, $\beta$ = 20 | 63 | 17 | 8 |
| Sobel, $\beta$ = 40 | 84 | 28 | 10 |
| Sobel, $\beta$ = 60 | 116 | 42 | 21 |
| Sobel, $\beta$ = 80 | 126 | 52 | 27 |
| Sobel, $\beta$ = 100 | 128 | 53 | 28 |
| TanTriggs, $\beta$ = 20 | 100 | 34 | 18 |
| TanTriggs, $\beta$ = 40 | 122 | 49 | 27 |
| TanTriggs, $\beta$ = 60 | 146 | 49 | 27 |
| TanTriggs, $\beta$ = 80 | 160 | 89 | 59 |
| TanTriggs, $\beta$ = 100 | **165** | **95** | **65** |

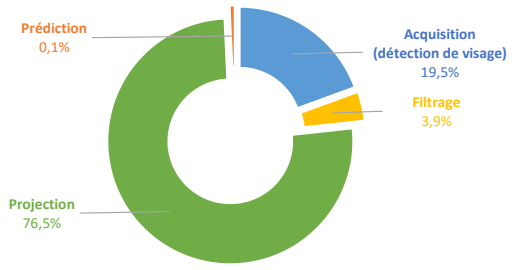
Table 6.13 – Temps de préparation des données pour une image : filtrage et projection dans l'espace de visage associé à la configuration des images d'entrée

Pour le Machine Learning classique, chaque image du visage détecté est filtrée puis projeté dans l'espace de visage. Une prédiction est effectuée à partir du vecteur résultant. D'après le tableau 6.13, répertoriant les temps de calculs du filtrage et de la projection d'un échantillon selon la configuration du système, et en considérant l'acquisition du visage (détection et extraction) pour chaque échantillon (40 ms), nous pouvons considérer que la préparation des données d'une image (acquisition, filtrage et projection) est bornée par 42 ms et 205 ms.
La table 6.14 révèle que, quels que soient le classifieur utilisé et la configuration d'entrée, la prédiction de classe d'un nouvel échantillon n'excède pas 200 $\mu$s. On remarque alors que la phase de préparation des données est prépondérante face à la prédiction. La répartition des temps de calcul de chaque étape dans le processus d'authentification, représentés sur la figure 6.6, montre que la durée de la phase de décision n'est pas significative (moins de 0,5% de la durée de l'authentification). La vitesse de traitement de la caméra intelligente dépend donc directement de la préparation des données. Ainsi, le système est capable d'effectuer une authentification de visage à un maximum de 24 ips (images par secondes) et à un minimum de 4,8 ips. Le Machine Learning classique répond donc aux contraintes de l'application, de par un fonctionnement que l'on peut considérer comme temps réel au regard d'une cadence vidéo standard à 25 ips. Cette vitesse d'authentification offre la possibilité de compenser les erreurs dues à l'orientation du visage ou à d'autres perturbations, simplement en effectuant une décision générale (vote à la majorité) à partir de multiples prédictions, légèrement espacées temporellement (non consécutives), sur quelques

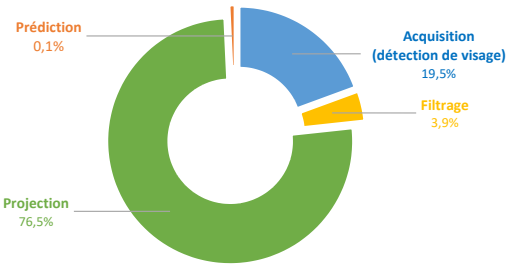
secondes d'acquisition.

| Filtrage | PCA 90% - [ms] | | PCA 70% - [ms] | | PCA 50% - [ms] | |
|---|---|---|---|---|---|---|
| | SVM | RF | SVM | RF | SVM | RF |
| Pas de filtrage | **0,11** | 0,12 | **0,06** | 0,11 | **0,05** | 0,10 |
| Sobel, $\beta$ = 20 | 0,12 | 0,12 | **0,06** | 0,08 | **0,05** | **0,07** |
| Sobel, $\beta$ = 40 | 0,14 | 0,12 | 0,07 | 0,11 | **0,05** | 0,09 |
| Sobel, $\beta$ = 60 | 0,16 | 0,15 | 0,09 | 0,10 | 0,07 | 0,11 |
| Sobel, $\beta$ = 80 | 0,17 | 0,14 | 0,10 | 0,11 | 0,08 | **0,15** |
| Sobel, $\beta$ = 100 | 0,16 | 0,13 | 0,10 | 0,12 | 0,08 | 0,12 |
| TanTriggs, $\beta$ = 20 | 0,13 | **0,11** | **0,06** | **0,07** | **0,05** | 0,08 |
| TanTriggs, $\beta$ = 40 | 0,14 | **0,11** | 0,09 | 0,10 | 0,06 | 0,11 |
| TanTriggs, $\beta$ = 60 | 0,18 | 0,13 | 0,10 | 0,10 | 0,07 | 0,10 |
| TanTriggs, $\beta$ = 80 | 0,19 | 0,14 | 0,13 | 0,14 | **0,10** | 0,11 |
| TanTriggs, $\beta$ = 100 | **0,20** | **0,19** | **0,14** | **0,16** | **0,10** | **0,15** |

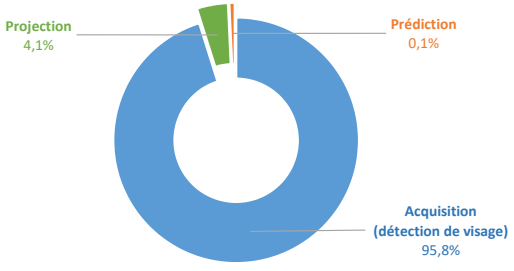
TABLE 6.14 – Temps de calcul de la prédiction de classe d'un échantillon pour les SVM et RF selon la contribution du filtrage et le seuil de l'ACP



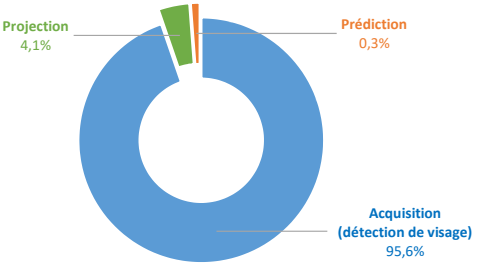

(a) SVM avec ACP 90%, TanTriggs 100%




(b) RF avec ACP 90%, TanTriggs 100%




(c) SVM avec ACP 50%, sans filtrage




(d) RF avec ACP 50%, sans filtrage

FIGURE 6.6 – Représentation graphique de la répartition des temps de calculs de la phase d'authentification pour les configurations les plus lentes et les plus rapides des SVM et des RF

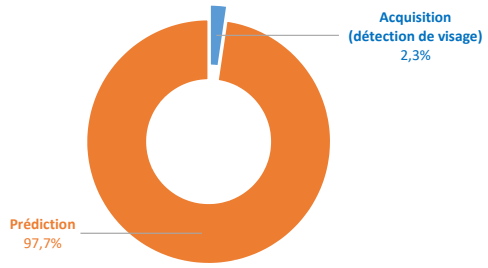
En fonctionnement réel, la phase complète d'authentification doit prendre en compte la lecture de la carte sans contact ainsi que le chargement du modèle biométrique par le système après sa décompression. Comme énoncé dans la section précédente (enrôlement), la lecture d'une carte MIFARE DESFire est d'environ 2,2 s pour un modèle de 7,5 Ko. Le système doit ensuite décompresser les données. Cette étape requiert au maximum 145 ms. Le chargement du modèle décompressé par le système est également rapide, entre 3 ms et 46 ms pour les SVM selon la configuration et entre 25 ms et 84 ms pour les RF.

La table 6.15 présente, pour chaque réseau CNN ré-entraînés, les temps de prédiction avec et sans quantification des réseaux. Systématiquement supérieur à une seconde, le temps de prédiction moyen des CNN est d'environ 1,8 s, soit environ neuf mille fois le temps de prédiction maximum du Machine Learning classique (SVM, ACP 90% et TanTriggs 100%). Tout comme pour l'apprentissage, on observe que la version 2 du réseau MobileNet est plus lente que sa version 1. On peut également considérer qu'il n'y a pas de différence entre les réseaux entraînés à partir de 1000 ou 40 images classes. En effet, la quantité de données dédiée à l'apprentissage de modifie pas l'architecture des CNN, et a seulement un impact sur la durée de l'apprentissage et les résultats d'authentification.

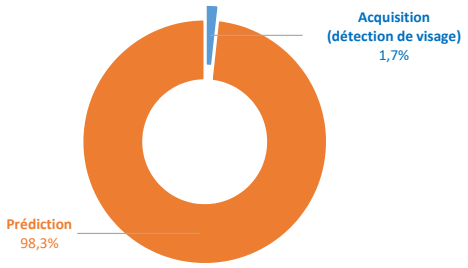
Avec les CNN, la phase de préparation des données consiste seulement à détecter le visage dans l'image afin de le transmettre au réseau. Cette fois ci, comme illustré par la figure 6.7, c'est le temps de préparation des données qui est négligeable en comparaison du temps de prédiction. La durée de l'authentification complète peut alors être simplifiée en ne tenant compte que des prédictions des CNN. Ainsi, la caméra intelligente est seulement capable d'effectuer une authentification de visage via des CNN à une vitesse inférieure à une image par seconde. La fréquence maximale étant de 0,73 ips, et de 0,44 ips pour la fréquence minimale. Par conséquent, même si le temps d'authentification de visages apparait comme encore acceptable (bien que lent) pour l'utilisateur, il devient difficile d'effectuer une authentification par décisions multiples sur une plage temporelle (afin de compenser les variations d'orientations du visage), cela augmentant considérablement le temps d'attente de l'usager.

| **Réseau** | **1000 images par class - [s]** | | **40 images par class - [s]** | |
|---|---|---|---|---|
| | Non Quant. | Quant. | Non Quant. | Quant. |
| v1_0.25_128 | **1,57** | **1,40** | **1,54** | **1,37** |
| v1_0.25_224 | 1,69 | 1,44 | 1,68 | 1,44 |
| v2_0.35_128 | 2,17 | 2,11 | 2,12 | 2,05 |
| v2_0.35_224 | **2,27** | **2,19** | **2,26** | **2,18** |

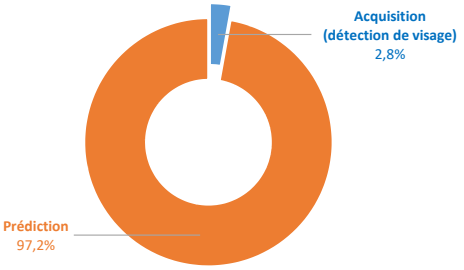
TABLE 6.15 – Temps de prédiction moyen pour un échantillon de MobileNet v1_025 et v2_035, non quantifiés et quantifiés, respectivement pour 1000 et 40 images d'apprentissage par classe



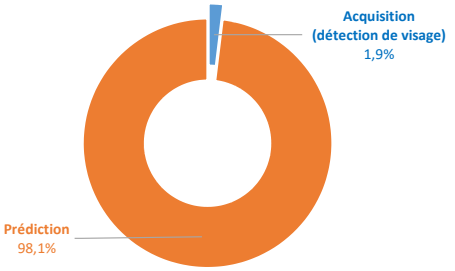

(a) MNet V1_025_224 N-Quant., 1000 images/classe




(b) MNet V2_035_224 N-Quant., 1000 images/classe




(c) MNet V1_025_128 Quant., 40 images/classe




(d) MNet V2_035_128 Quant., 40 images/classe

FIGURE 6.7 – Répartition des temps de calculs de la phase d'authentification pour les configurations les plus lentes et les plus rapides des réseaux MobileNet v1_025 et v2_035

#### 6.2.2.3/ DISCUSSION

Étant donné que les caractéristiques des visages sont sujettes à des variations au fil du temps, une mise à jour régulière des modèles est requise. Cependant les modèles issus des SVM et RF ne peuvent pas être directement mis à jour. De plus, aucune donnée personnelle du sujet ne reste sauvegardée dans le système. Toutefois, les temps d'apprentissage relativement courts de ces méthodes permettent de résoudre ce problème. En effet, si l'authentification est un succès, le système peut utiliser les images acquises et validées, qui ne seront conservées que sur la durée de ce processus. Celles-ci permettent alors d'effectuer un nouvel apprentissage, remplaçant ensuite l'ancien modèle sur la carte. Pour les configurations les plus rapides, cette mise à jour peut être inaperçue. Nous avons pu observer qu'une carte Raspberry PI 3 est capable d'effectuer l'apprentissage d'un CNN "léger" par transfert de connaissances et de l'utiliser par la suite. Cependant, la puissance limitée de cette plateforme n'est pas suffisante pour un fonctionnement temps réel stricte. Pour cette étude, nous avons choisi d'utiliser des CNN pour toute la chaîne de traitement (extraction de caractéristiques et classification). Cependant, dans la littérature, les CNN sont souvent utilisés comme seuls extracteurs de caractéristiques. La classification est alors réalisée par un algorithme du Machine Learning classique, alimenté par les données fournies par le CNN. Cette utilisation est donc proche de notre implémentation du Machine Learning traditionnel où, par analogie, l'ACP rempli la même fonction que le CNN. Cependant, même dans ce cas, une telle utilisation ne donnerait pas nécessairement de bien meilleurs temps de calculs. La partie extraction de caractéristiques du CNN devant toujours être utilisée, le temps de préparation des données d'un échantillon serait toujours supérieur à l'extraction de l'ACP. De plus, la dimension du vecteur en sortie des couches de convolution (vecteur de 1024 éléments pour MobileNet v1), augmentera nécessairement la taille de stockage du modèle.

La table 6.16 présente un résumé des temps de calcul des traitements de la phase d'authentification pour la configuration retenue, utilisée pour la fusion de décision. Pour rappel, cette configuration consiste en un filtrage de TanTriggs en guise de prétraitements, une extraction de caractéristiques avec l'ACP dont l'espace de visage a été réduit à 50% de l'inertie de ses valeurs propres, et une classification avec un SVM. Comme décrit précédemment, la décision représente une infime partie du temps requis pour effectuer une authentification à partir d'une image. Avec 58 millisecondes, la configuration sélectionnée permet à la caméra intelligente d'effectuer une authentification du visage à une fréquence de 17 images par secondes.

| Traitement | Temps de calcul |
|---|---|
| **Acquisition [ms]** | 40 |
| **Prétraitements [ms]** | 18 |
| **Décision [ms]** | 0,05 |
| **Authentification [ms]** | 58,05 |
| **fréquence [ips]** | 17 |

TABLE 6.16 – Temps de calculs de la phase d'authentification du visage, avec une image, pour la configuration retenue : filtre de TanTriggs avec $\beta = 20$, ACP 50% et classification SVM

### 6.2.3/ AUTHENTIFICATIONS DU DOIGT - PROTOTYPAGE D'UN SYSTÈME D'ACQUISITION

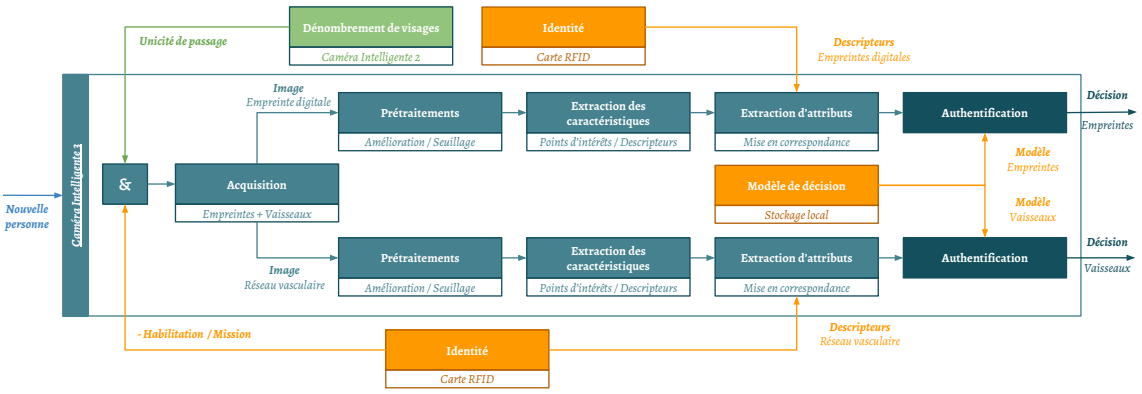



FIGURE 6.8 – Protocole de l'authentification des deux modalités du doigt

Le figure 6.8 illustre le protocole d'authentification par les empreintes digitales et le réseau vasculaire du doigt. Étant situés sur la même zone du corps humain, leur acquisition est vouée à être effectuée par un unique système sans contact. Contrairement au visage, ces deux modalités n'ont pas vocations à être traitées par une caméra intelligente basée sur une carte Raspberry. Ici, nous proposons un système d'acquisition permettant d'obtenir la mesure de ces deux modalités par la rotation d'une caméra autour du doigt. Cette section présente donc le concept et la conception d'un prototype de scanner rotatif.

#### 6.2.3.1/ Présentation du système

**Limitation des acquisitions sans contact**

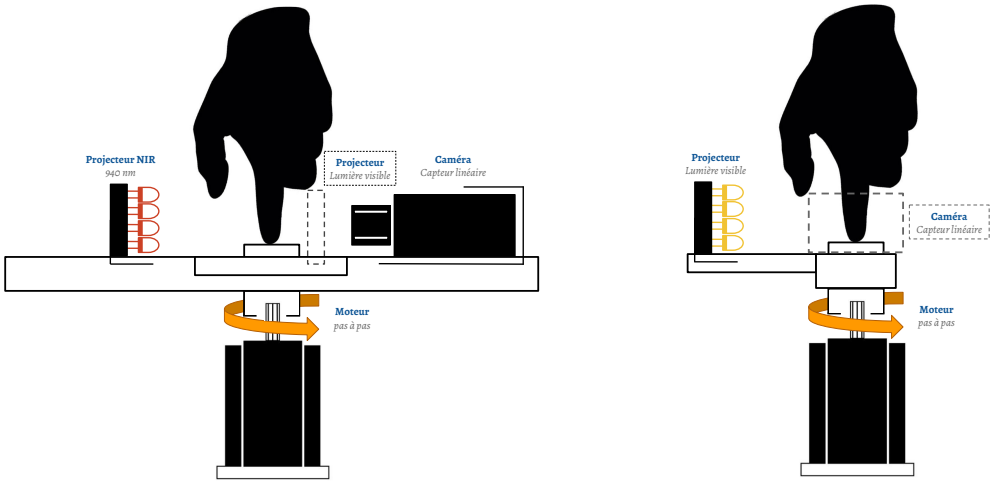
Les empreintes digitales et le réseau vasculaire du doigt peuvent être acquis par un même système. Les systèmes existants permettent l'acquisition simultanée de chacune d'entre elle sur une phalange différente (veines acquises sur la deuxième phalange). Cependant, bien que ces systèmes contraignent le positionnement du doigt, de légères rotations et/ou translations peuvent être observées entre deux acquisitions pour une même personne. Principalement pour les empreintes digitales, ces rotations peuvent conduire à un changement radical de la zone observée.
De plus, contrairement aux systèmes d'acquisition avec contact, la capture sans contact des empreintes digitales ne permet pas la mesure de la conductivité du doigt. Il est par conséquent plus aisé de reproduire cette modalité à partir de "traces" laissées par le propriétaire de ces empreintes et de contourner le système grâce à n'importe quel matériau représentant ces fausses empreintes. Pour le réseau vasculaire, la copie est plus complexe du fait de l'emplacement sous-cutané de cette modalité, nécessitant alors un procédé d'acquisition particulier.

Nous proposons alors d'étudier un système d'acquisition original de ces deux modalités, permettant de contourner les limitations des systèmes traditionnels. Ce prototype, conçu au sein de notre laboratoire, se présente comme un scanner rotatif permettant d'acquérir ces deux modalités sur le tour du doigt. Ainsi, les variations en rotation et en translation sont en grande partie évitées. De plus, la copie est bien moins triviale et nécessite l'accès au doigt complet de l'individu habilité dont on souhaite usurper l'identité. De plus, il a été montré que la face arrière du doigt, de par la forme des plis de la peau, fourni autant d'informations que la face avant où se trouvent les empreintes [239].

**Principe général du système proposé**

Ce système, dont le schéma de principe est illustré par la figure 6.9, se présente donc comme un scanner rotatif du doigt. Fixée sur un rail entraîné en rotation par un moteur pas à pas, une caméra linéaire acquiert rapidement plusieurs profils de la périphérie du doigt. Ce dispositif est muni de deux projecteurs leds, pointés sur le doigt, diffusant différentes longueurs d'ondes. Un premier projecteur est utilisé à l'arrière du doigt (visible dans le plan de la figure 6.9a). Celui-ci émet une lumière dans le proche infrarouge (à 940 nm) afin d'acquérir les veines par transmission. Un second projecteur, fixé latéralement à quelques degrés de la caméra (visible dans le plan de la figure 6.9b), illumine le doigt d'un éclairage "rasant" dans le domaine visible afin d'acquérir les empreintes digitales par réflexion.
Après la rotation de la caméra sur le tour du doigt, le système doit retourner à sa position d'origine par une rotation dans le sens contraire. Nous exploitons cette particularité afin d'obtenir une seule de ces modalités par rotation. Cela est effectué en allumant seulement le projecteur de lumière visible à l'aller, puis seulement le projecteur infrarouge au retour de la caméra. Deux images sont alors acquises : une image de chaque modalité mise "à plat" sur le tour du doigt.

Dans la suite de ce chapitre, les différentes phases de l'élaboration de prototype sont présentées. D'une part, une validation du principe d'acquisition des deux modalités par rotation est effectuée par simulation. Un logiciel de modélisation 3D est utilisé afin de simuler le fonctionnement du système et d'observer les images pouvant potentiellement être acquises. D'autre part, la phase de conception du prototype est présentée, incluant les choix des divers composants. Pour finir, nous illustrons le fonctionnement de ce prototype par les premières images réelles acquises.



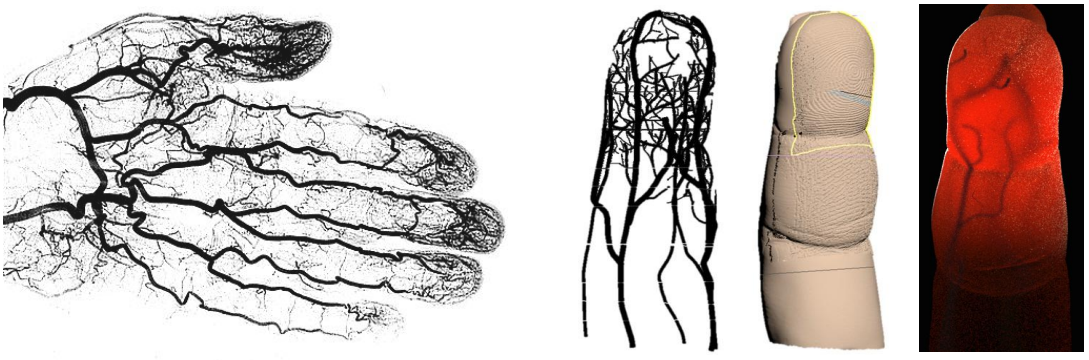

(a) Vue de face (b) Vue de profil

FIGURE 6.9 – Schéma de principe du prototype de scanner rotatif

### 6.2.3.2/ ÉLABORATION D'UN PROTOTYPE

**Validation du principe par simulation**

Afin de valider le principe du système et éventuellement y apporter des modifications pour ensuite élaborer un prototype, nous l'avons modélisé dans un logiciel de création 3D (3DSMax). En se référant à la répartition du réseau vasculaire dans les doigts, figure 6.10a, nous avons créé un modèle de réseau vasculaire similaire, figure 6.10b, avec des vaisseaux plus fins et plus nombreux aux extrémités du doigt. Différentes couches de textures en relief, issues d'une photo de haute qualité, ont été apposées sur le tour du doigt, lequel a été modélisé autour de ce réseau vasculaire. Elles comprennent les empreintes digitales mais également les plis de la peau sur le tour du doigt.



(a) Réseau vasculaire de la main [3] (b) Simulation sous 3DSMax

FIGURE 6.10 – Simulation 3D d'un doigt avec ses veines et son empreinte digitale

3. Image source : volusiahandsurgery.com/the-cold-hand/

À partir du principe du système proposé, nous définissons la trajectoire de la caméra virtuelle et des éclairages simulés à partir d'un cercle autour du doigt. La cible de la caméra est donc fixe et centrée sur le doigt. La trajectoire des éclairages a également été paramétrée de manière à conserver une orientation vers le doigt, lequel demeure fixe dans l'espace. De plus, la position de chaque élément du système reste fixe par rapport à la position des autres. Une modélisation complète du système de scanner rotatif est donc obtenue, illustrée figure 6.11.
Nous avons paramétré la caméra afin d'observer une représentation de la zone d'intérêt du doigt occupant la majeure partie de l'image et ainsi obtenir un résultat suffisamment résolu au niveau des empreintes digitales. Cette région d'intérêt est composée des deux premières phalanges. Les matériaux constituant le doigt ont par la suite été configurés expérimentalement pour simuler les différentes interactions entre la lumière et les tissus ainsi que les veines, telles que la transmission, la diffusion ou l'absorption de la lumière. Un aperçu du rendu des vaisseaux sanguins par éclairage à l'arrière du doigt est présenté figure 6.10b.

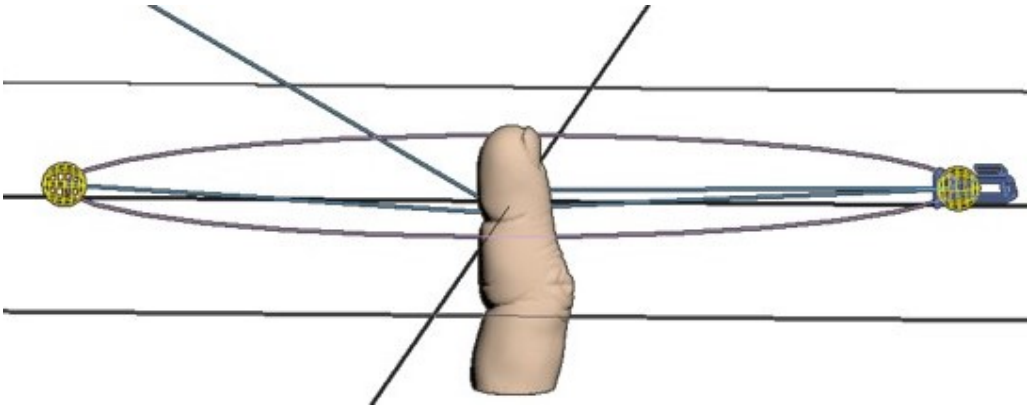
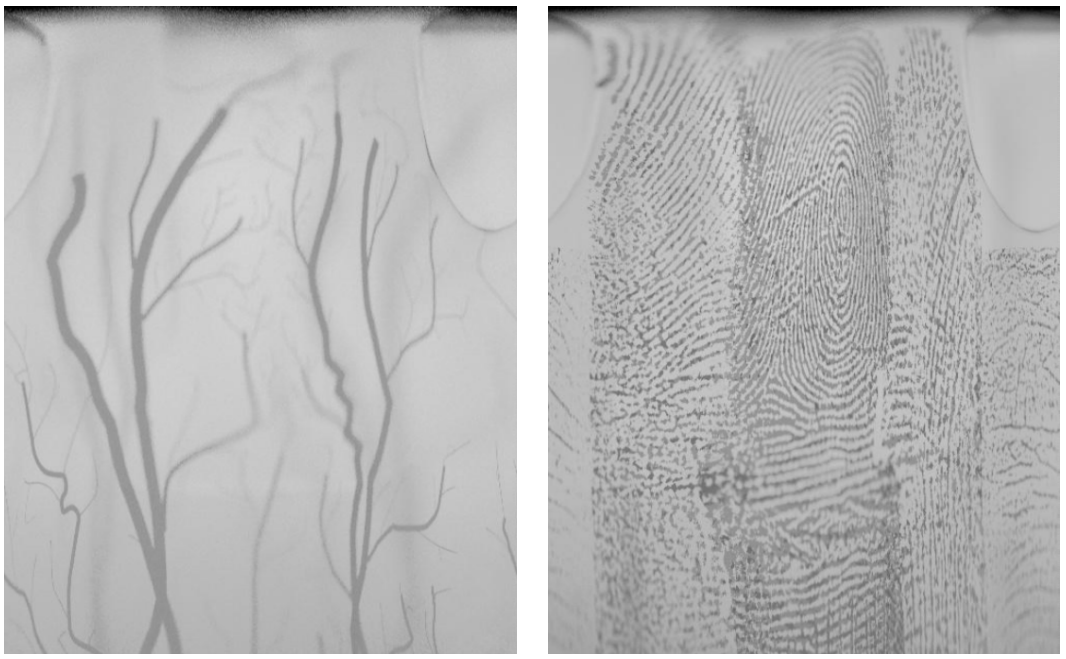


FIGURE 6.11 – Simulation 3D du scénario d'acquisition du système rotatif

Lors de la simulation, la caméra est déplacée le long de sa trajectoire, toujours orientée vers le doigt, et une simulation de la scène est effectuée à chaque déplacement. La colonne de pixels centrale de chaque image obtenue est ensuite extraite. Ces colonnes, mises côte à côte, composent alors une nouvelle image représentant une reconstruction aplanie du tour du doigt. Dans le but d'obtenir une image reconstituée suffisamment résolue, nous avons fixé à 500 le nombre d'images générées. Une simulation de la scène est donc effectuée à chaque déplacement de $0,72$ degré.

Un exemple de reconstitution des veines et des empreintes est présenté en figure 6.12. Sur l'image des vaisseaux reconstitués, figure 6.12a, on peut observer que les structures plus profondes apparaissent plus floues (ou diffusent) que les structures proches de la surface. Du fait d'une certaine difficulté à apposer les textures de la peau sur la modélisation du doigt, le résultat de la reconstitution des empreintes digitales (figure 6.12b) n'est pas aussi net. On observe en effet les trois modèles de textures utilisés. Bien que cette simulation soit une modélisation simpliste des tissus ainsi que des interactions entre la lumière et ces derniers, elle permet de valider le principe du système. L'observation des deux images obtenues confirme alors une robustesse accrue aux variations de positions du doigt entre différentes acquisitions. Le principe étant validé, le développement d'un prototype a donc pu être réalisé.



(a) (b)

FIGURE 6.12 – Reconstitutions aplanies des rendus noirs et blanc des veines (a)et des empreintes (b), obtenues par simulation du système sous le logiciel d'édition 3DSMax

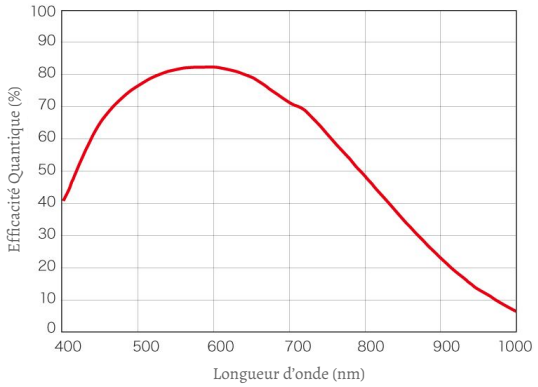
### Mise en œuvre matérielle

Différentes catégories d'éléments composent ce prototype. Une partie mécanique est formée des pièces telles que le rail et les supports de la caméra et des éclairages sur lesquels ils sont fixés. À des fins d'expérimentations, ces supports ont été conçus de façon à être mobiles sur le rail avec une position réglable. La partie électronique permet de mettre en rotation le rail. Elle comprend un moteur pas à pas pouvant être paramétré en micro-pas, une carte de puissance contrôlant le moteur, un capteur d'angle mesurant la position du moteur (et donc son déplacement angulaire) et une carte de commande du moteur. La partie optique est constituée de la caméra, de son objectif et des deux éclairages visible et proche infrarouge.
Comme l'étude algorithmique, la phase d'élaboration de ce prototype doit répondre à un cahier des charges. Celui-ci est régi notamment par les spécifications techniques de ses éléments (caméra, éclairage, moteur), ainsi que par ses objectifs principaux et des contraintes liées à sa phase d'utilisation. Par conséquent, une étude technique est requise, suivie par la conception du système (forme, dimensions, etc) et enfin sa fabrication. Ces contraintes sont entre autres le placement des câbles afin de ne pas gêner la rotation, la vitesse de rotation du rail ou encore la taille du système. En effet, ce dernier ne doit pas être d'une envergure trop importante ce qui rendrait son intégration trop complexe et requerrait un couple moteur important du fait de l'inertie du rail en rotation.

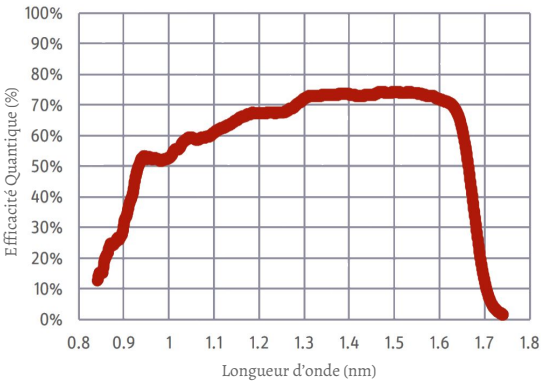
Afin de déterminer la taille du système, la caméra doit être sélectionnée au préalable. En effet, de nombreux paramètres dépendent du choix de la caméra et de son optique. La taille de la caméra détermine la taille de son support. Ses caractéristiques optiques et son objectif déterminent la distance de mise au point (distance focale) et donc l'envergure du système (longueur du rail). De plus, le poids de la caméra joue également un rôle dans la détermination de la puissance du moteur nécessaire pour permettre une rotation fluide (sans saccades) du système à une vitesse convenable.

Les différentes interactions de la lumière infrarouge et des tissus rendent difficile l'observation des vaisseaux vasculaires. Comme énoncé dans le chapitre 5, les images résultantes sont en effet faiblement contrastées et les vaisseaux peu distincts. De plus, les longueurs d'ondes optimales, situées entre 800 µm et 1000 µm, correspondent au rendement quantique (ou efficacité quantique : *QE*) le plus faible des capteurs sCMOS (*"CMOS scientifique"* comme le montre la figure 6.13a.
En comparaison, la plage de fonctionnement des capteurs "InGAas" (Arséniure de gallium et d'indium) se trouve dans le proche infrarouge et l'infrarouge moyen. Comme le montre la figure 6.13b, son rendement quantique est également faible (15%) à 850 nm, mais son augmentation rapide donne un QE de 50% à 940 nm. Par conséquent, une caméra InGaAs semble présenter un certain avantage, comparé aux caméras sCMOS, dans les longueurs d'ondes permettant l'observation des vaisseaux sanguins.

Notre système n'utilisant qu'une colonne de pixels afin de reconstituer l'image du tour du doigt, et les veines étant la modalité la plus complexe à acquérir, nous nous sommes orientés vers une caméra linéaire InGaAs. Le capteur de cette caméra est composé d'une ligne de 1024 pixels de taille $12,5\times 12,5$ µm. Elle permet de traiter des images en niveaux de gris de profondeur 14 bits à une fréquence maximale de 40000 profils par seconde. Cette fréquence est néanmoins dépendante du temps d'exposition, lequel pouvant être défini entre 3 µs et quelques secondes.
L'optique sélectionnée est dédiée à des applications de vision industrielle. De dimensions $6,6\times 5,1$ cm (diamètre $\times$ longueur), elle dispose d'une ouverture réglable manuellement de $f/1.4$ à $f/22$. Sa distance minimale de mise au point est de 10 cm pour une longueur focale de 12,5 mm, avec un angle de champ horizontal de 54° et un angle de champ vertical de 42°.




(a) Rendement Quantique d'un capteur sCMOS[4]




(b) Rendement Quantique du capteur utilisé[5]

FIGURE 6.13 – Comparaison des rendements quantiques de capteurs selon la longueur d'onde

Le poids de la caméra sélectionnée est de 200 grammes et celui de l'optique choisie est d'environ 300 grammes. La longueur de l'ensemble formé par la caméra et son optique est de 14 cm et la distance minimale de mise au point de cet objectif est de 10 cm. En l'état, cela implique un rail d'une longueur d'au moins 25 cm du côté de la caméra, conduisant à un système dont les dimensions sont bien trop importantes (50 cm d'envergure). De plus, à cette distance, la taille de l'observation du doigt est bien trop petite pour pouvoir être exploitée, et le poids de la partie optique située à l'extrémité du rail conduit à une trop grande inertie de mouvement pour garantir une rotation fluide. Par conséquent, nous faisons usage d'une bague allonge, permettant alors de positionner le couple caméra et son optique à une distance de 4 cm du doigt. Ce gain comporte une contrepartie. En effet, en réduisant la distance focale par une bague allonge, on réduit la plage

4. Image source : Capteur sCMOS "C11440-42U30" - HAMAMATSU
5. Image source : Documentation technique Caméra "Xenics Lynx 1024 GE"

de mise au point. De plus, l'ouverture utilisée afin d'obtenir plus de lumière réduit la profondeur de champ. Avec une profondeur de champ très courte, comme le doigt n'est pas parfaitement cylindrique, il est difficile d'obtenir une acquisition nette sur tout le tour du doigt. Un compromis est donc à effectuer.

Les leds constituant le système d'éclairage ont soigneusement été sélectionnées. Étant donné la plage de sensibilité du capteur choisi, les deux projecteurs utilisés doivent être composés de leds infrarouges. Chaque projecteur est formé de 6 leds disposées verticalement de façon à éclairer le doigt dans sa longueur. Chacune dispose d'un angle d'émission restreint afin de focaliser la puissance lumineuse sur l'axe central du doigt.
Le projecteur éclairant les empreintes digitales, disposé latéralement afin d'avoir un angle d'incidence rasant, est composé de leds émettant une longueur d'onde de 860 nm. L'intensité énergétique de son émission est de 700 $mW/sr$ avec un cône d'émission d'angle 6°. Afin de ne pas surexposer le doigt, l'intensité lumineuse de l'éclairage rasant n'est pas utilisée à son maximum lors de son fonctionnement. Les leds permettant d'observer les veines par transmission projettent une longueur d'onde de 950 nm et sont positionnées à l'opposé de la caméra. L'intensité énergétique de son émission est de 600 $mW/sr$ pour un cône d'intensité d'angle 6°.

Le moteur sélectionné permet une rotation à un pas de $1,8°$, pouvant être divisé en micro-pas (jusqu'à 256 micro-pas pour un pas). Nous avons sélectionné un micro pas de $0,18$, permettant une mise en rotation fluide du système sans secousses au démarrage, et une précision suffisante pour notre application. L'utilisation du moteur en mode micro-pas résulte en une réduction du couple, d'environ 30% du couple du pas complet. Par conséquent, le moteur sélectionné dispose d'un couple de 500 mNm. Afin de mesurer les déplacements du moteur et permettre le déclenchement d'une acquisition à chaque déplacement de $0,72°$, un capteur d'angle est placé sous l'arbre du moteur. Celui-ci est relié à la carte de commande du moteur. Celle-ci est une carte arduino, reliée à une station de travail (ordinateur). Cette carte arduino commande la carte de puissance du moteur, permettant de sélectionner se vitesse de déplacement ainsi son pas. Nous avons défini la vitesse de rotation du moteur de telle sorte qu'une rotation soit effectuée en 2,5 secondes. Ainsi, afin d'obtenir 500 acquisitions pour une rotation complète, le temps d'intégration de la caméra peut être ajusté jusqu'à un maximum de 5 millisecondes. L'acquisition des deux modalités en question nécessite alors une durée de 5 secondes (acquisition des empreintes digitales à l'aller et acquisition des vaisseaux au retour du rail).

Dans le but de valider l'aspect général du système, nous avons réalisé un modèle en trois dimensions, proche du système final, à l'aide du logiciel Sketchup. La figure 6.14 présente un aperçu de cette modélisation. On peut y observer le système monté et composé d'un support pour la rotation du rail. De plus, nous proposons une version en fonctionnement, confinée dans une boite tel que pourra l'être le système final. Par la suite, cette modélisation a servi de base à un modèle corrigé et réaliste, conçu par CAO (Conception Assistée par Ordinateur). Ce dernier modèle a permis l'usinage des différentes pièces du prototype. Le système assemblé et en fonctionnement est illustré par la figure 6.15. Le rail, la colonne du moteur ainsi que le bloc support du système ont été usinés avec de l'aluminium. Les pièces amovibles telles que supports d'éclairage et de la caméra sont en plastique, fabriqués par imprimante 3D.

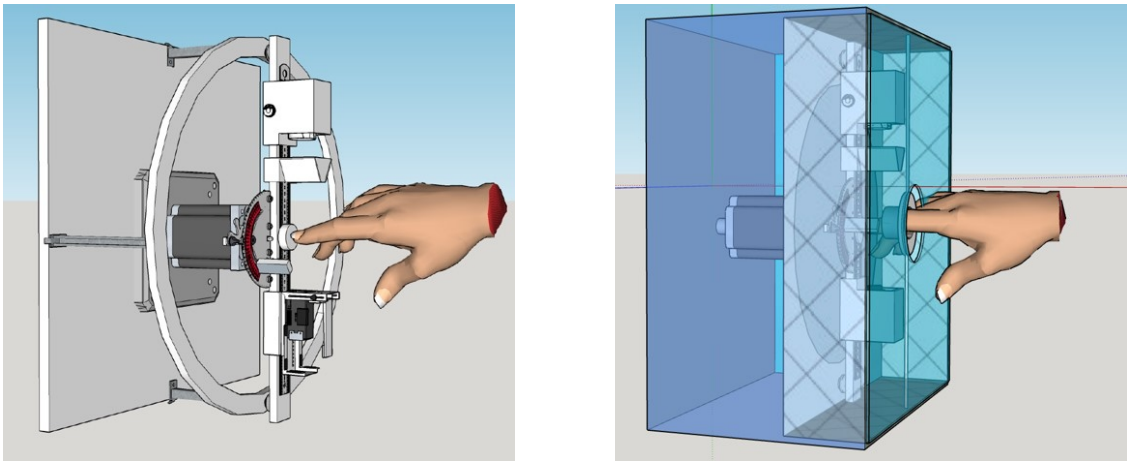


FIGURE 6.14 – Réalisation 3D du prototype et son boitier sous Sketchup

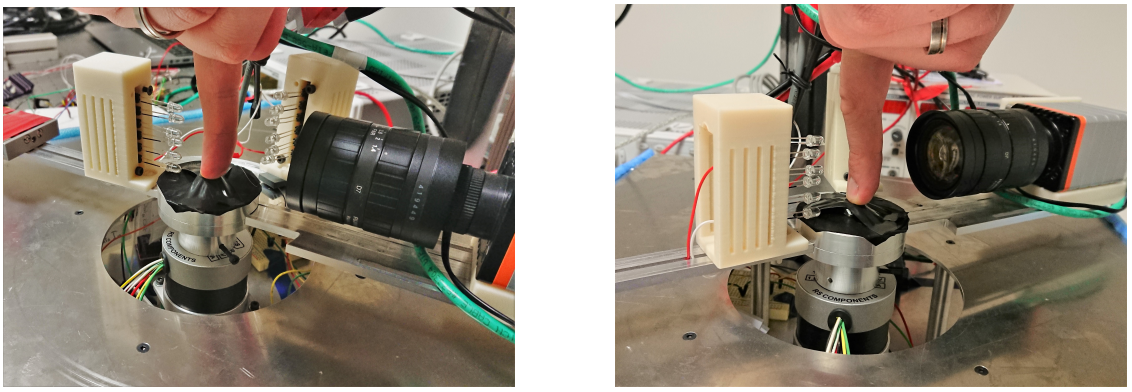


FIGURE 6.15 – Première version du prototype

### Premières acquisitions

Le prototype conçu étant fonctionnel, nous avons pu obtenir des premières images après un paramétrage expérimental du système. Les images de la figure 6.16 6.17 représentent les acquisitions des empreintes digitales de deux personnes distinctes, obtenues avec l'éclairage à 860 nm. On peut observer les motifs bien définis des empreintes et des structures formées par les plis de la peau. Le paramétrage n'est cependant pas optimal du fait d'un contraste non uniforme et de certains problèmes de reconstruction du doigt notamment en bordure des images.

Avec une caméra linéaire, l'image reconstituée n'est observable qu'après l'acquisition de tous les profils du tour du doigt. Par conséquent, à partir d'une simple colonne de pixels, le paramétrage des éléments du système n'est pas trivial. Cela rend particulièrement difficile l'acquisition des vaisseaux sanguins, présentés en figure 6.17. Sur ces images, on distingue certains vaisseaux mais le résultat n'est pas celui attendu. Le contraste de l'acquisition étant faible, les paramètres décisifs sont liés à la puissance du projecteur, à l'ouverture de l'objectif, au temps d'exposition de la caméra ainsi qu'à la vitesse de rotation du système.

La profondeur de champ étant réduite par la bague allonge, il est nécessaire de maintenir une ouverture faible de l'objectif afin de ne pas réduire d'avantage la plage de mise au point. Par conséquent, le temps d'intégration de la caméra doit nécessairement être augmenté, et la vitesse du moteur adaptée.

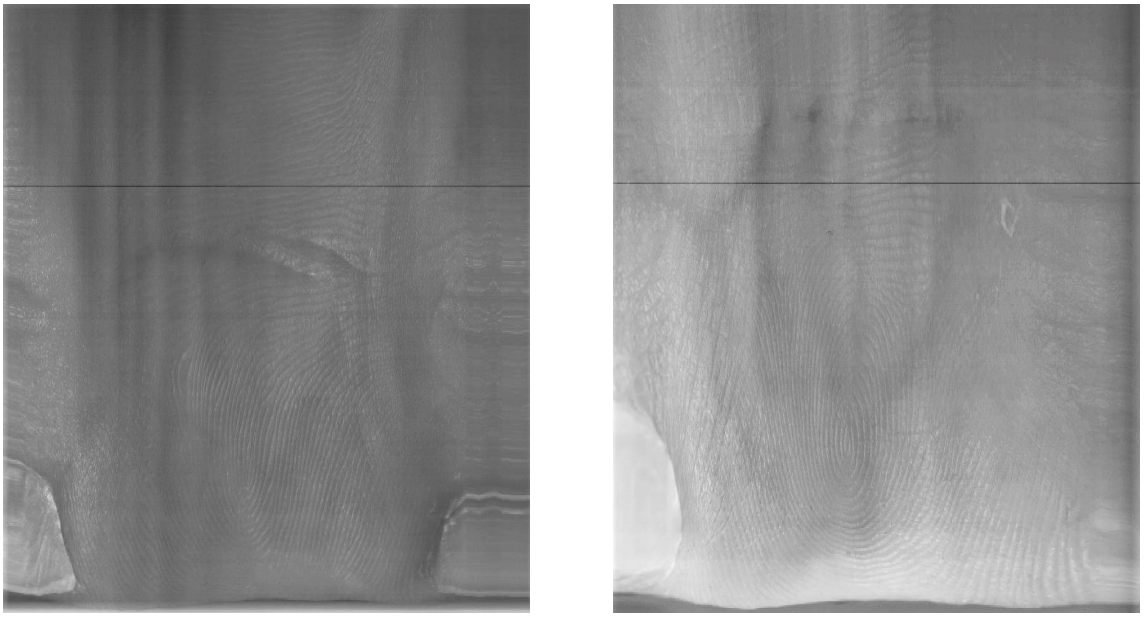


FIGURE 6.16 – Exemple d'acquisition d'empreintes digitales de deux personnes distinctes avec le prototype du scanner rotatif

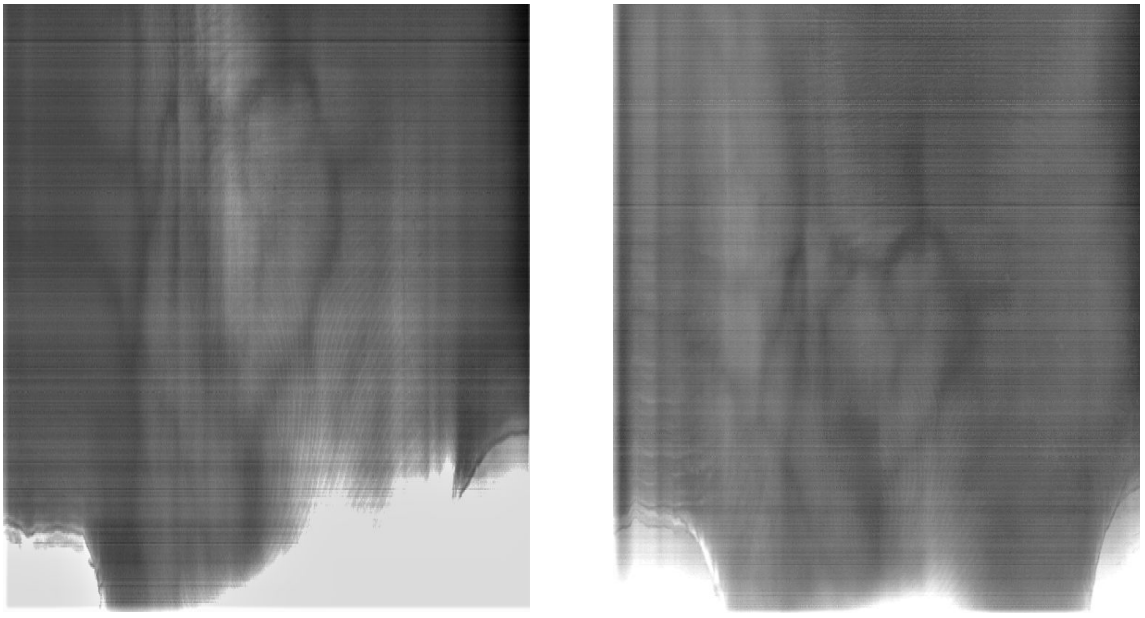


FIGURE 6.17 – Exemple d'acquisition du réseau vasculaire de deux personnes distinctes avec le prototype du scanner rotatif

Hormis un réglage optimal du système, il est possible d'agir sur d'autres paramètres afin d'obtenir de meilleures images. Les détecteurs en général (comme tous les composants électroniques actifs), et particulièrement les InGaAs, génèrent des charges par agitation thermique. C'est à dire que plus la température est élevée, plus le nombre de charges est important. Par conséquent, les capteurs InGaAs sont sensibles à la température de fonctionnement de la caméra. Sur la figure 6.18, on observe le rendement quantique d'un capteur InGaAs selon différentes longueurs d'ondes de l'infrarouge pour différentes températures de de -133°C (140 Kelvin) à 27°C (300 Kelvins). Il en ressort que les meilleurs rendements dans les faibles longueurs d'ondes sont obtenues pour les températures les plus basses.

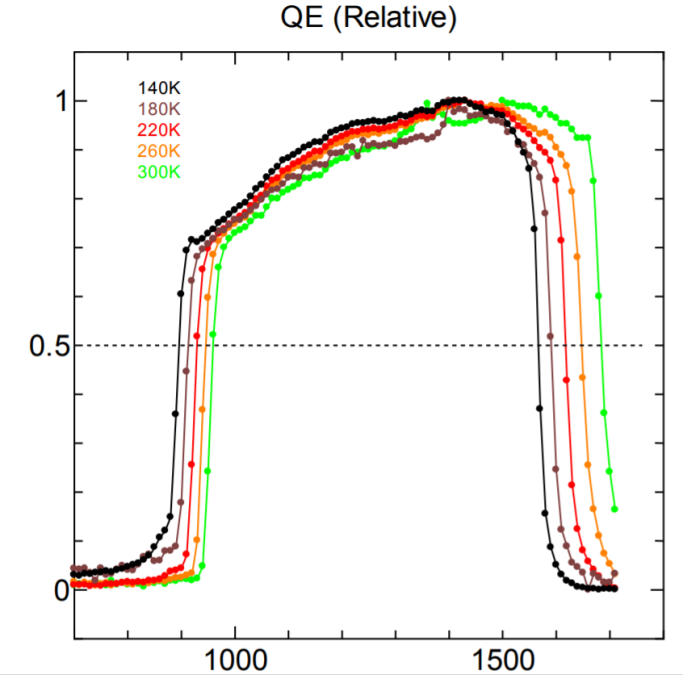



FIGURE 6.18 – Sensibilité d'un capteur InGaAs selon différentes températures en Kelvin [281]

De plus, dans le cas d'un capteur d'images en mode intégration (acquisition d'une image par accumulation de photons sur une certaine période de temps), les charges induites par l'agitation thermique s'accumulent pendant les poses au même titre que celles issues du signal lumineux utile. Ce qui résulte, pour des températures de fonctionnement proches de la température ambiante (25°C), l'image acquise comporte une grande proportion de bruit. Grâce à un système dédié à des expérimentations en spectroscopie, notamment pour une application *"mixing wave swir-terahertz"* [282, 283], nous avons vérifié expérimentalement l'influence de la température sur un capteur InGaAs. Ce système, composé d'une caméra InGaAs refroidie par "module Peltier" couplé à un système de refroidissement liquide, a été fourni par David Darson, ingénieur de recherche au laboratoire de physique de l'École normale supérieure de Paris (LPENS).
Les images résultantes, figure 6.19 et 6.20, illustrent deux cas de figure pour lesquels la température de fonctionnement a été progressivement réduite. Chaque figure représente une certaine scène éclairée par un projecteur de longueur d'onde 950nm. Pour la première figure, l'intensité du projecteur est faible, l'objectif est fermé au maximum et donc la scène a été acquise par une intégration sur 2 secondes. La température de fonctionnement a été progressivement réduite de 25°C à -196°C. Pour la seconde figure, avec une intégration de 100 millisecondes, l'intensité du projecteur est plus forte et l'objectif plus ouvert. La température varie de 25°C à -10°C. Avec une scène très faiblement éclairée, l'intégration sur 2s à température ambiante montre une forte accumulation de

charge au travers d'une image blanche. Ce n'est qu'à partir de -100°C que le bruit est suffisamment réduit pour pouvoir observer la scène. Cependant, la meilleure image est obtenue pour une température proche de -200°C, pour laquelle le contraste est maximum et le bruit est minimisé. Avec un temps d'intégration beaucoup plus court, l'acquisition à la température ambiante permet d'observer la scène mais le contraste est faible. Comme précédemment, la réduction de température de fonctionnement du capteur permet d'augmenter le contraste de l'image. On remarque donc qu'en fonctionnement dans des températures négatives, le capteur montre une plus grande sensibilité à la lumière et une plus grande résistance aux bruits électronique et d'acquisition (accumulation de charges), ce qui permet alors de fournir des images plus contrastées et de meilleure qualité.

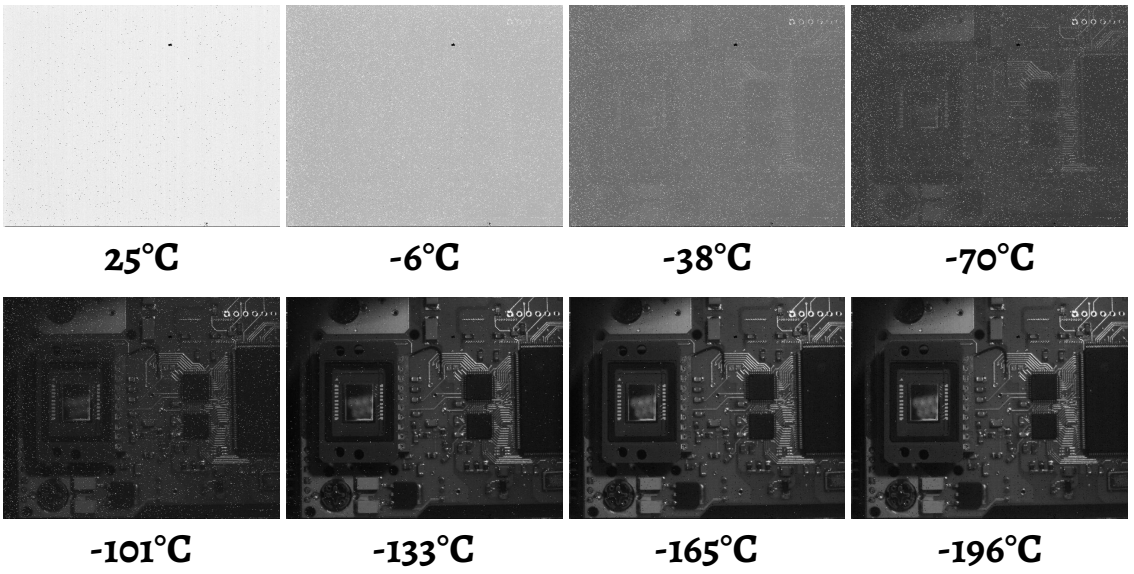



FIGURE 6.19 – Acquisition d'une scène sombre par un capteur InGaAs, soumis à différentes températures, avec 2 secondes d'accumulation

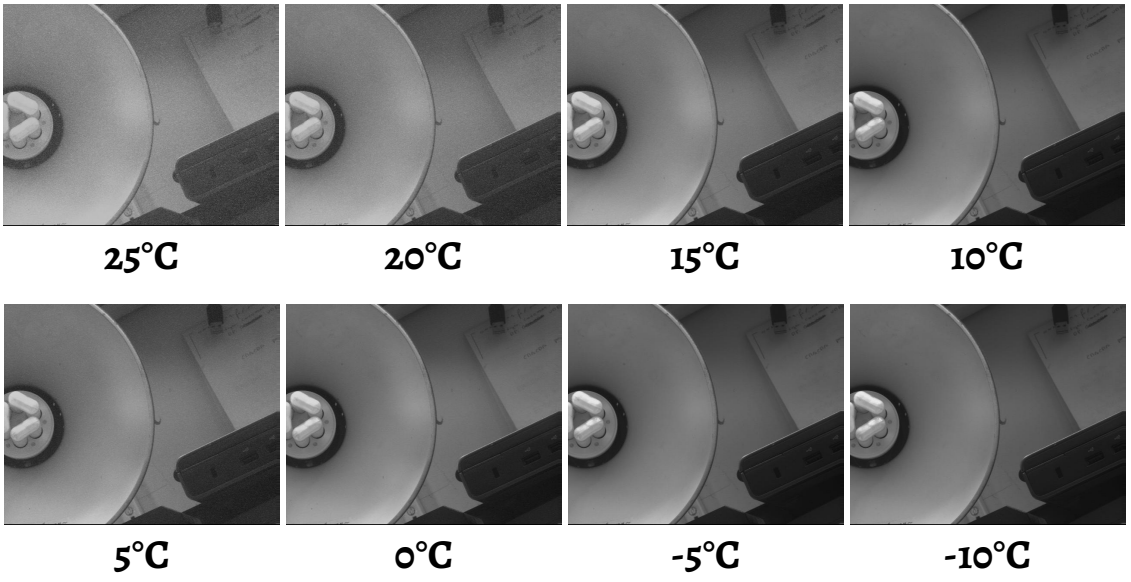



FIGURE 6.20 – Acquisition d'une scène claire par un capteur InGaAs, soumis à différentes températures, avec 100 millisecondes d'accumulation

Au sein de notre laboratoire, avec ce même système, nous avons reproduit l'expérience avec l'acquisition de la première phalange d'un doigt, dans un caisson privé de lumière extérieure. Le module de refroidissement a maintenu la température de la caméra à -40°C. De par un éclairage au dos du doigt par un projecteur infrarouge de longueur d'onde 950 nm, nous avons été en mesure d'observer simultanément les empreintes digitales et le réseau vasculaire. Comme le montre la figure 6.21, les vaisseaux captés sont très fins, et les empreintes digitales sont visibles. Ce système de refroidissement présente donc un avantage considérable. Mais son intégration à notre prototype semble complexe. En effet, ce genre de système est très imposant, et ne permet une grande souplesse dans les mouvements requis par notre scanner rotatif.

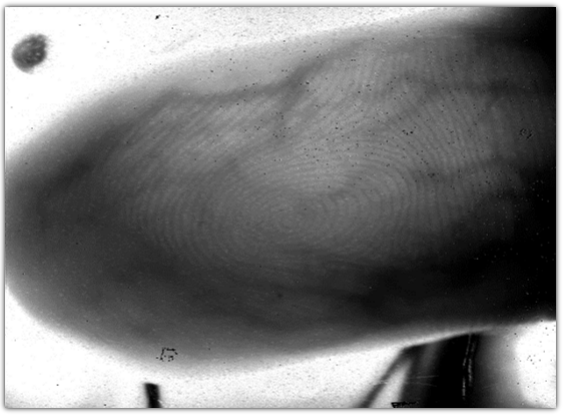


FIGURE 6.21 – Acquisition du doigt, éclairé à l'arrière par un projecteur proche infrarouge, par une caméra fixe composée d'un capteur InGaAs refroidit à -40°C

Malgré ces tests préliminaires encourageants au niveau de la génération des premières images, le prototype basé sur un capteur linéaire ne permet pas à l'heure actuelle d'acquérir des images d'une qualité suffisante pour réaliser les traitements des deux modalités considérées au niveau du doigt. Cependant, cette solution est présentée ici comme une base pour de futurs développements qui nous apparaissent alors prometteurs dans la continuité des travaux réalisés.

# 7

# Conclusion et perspectives

Cette thèse a pour vocation de répondre à une problématique de sécurisation d'accès à des zones restreintes, mêlant la biométrie multimodale et l'apport de la vision assistée par ordinateur au travers notamment de l'intelligence artificielle. De plus, les travaux de cette thèse s'inscrivent dans le cadre d'un projet industriel plus vaste. Ainsi, au-delà d'une réflexion purement théorique, la recherche entreprise dans ces travaux a été fortement axée sur des aspects d'adéquation algorithme architecture afin d'établir un compromis entre les algorithmes développés et l'implantation matérielle nécessaire au déploiement d'une telle solution. Le cahier des charges encadrant les développements mis en place porte notamment sur la capacité du système à garantir la vie privée des usagers tout en proposant un système de sécurité biométrique permettant une authentification rapide et fiable (soit sa faculté à détecter une intrusion ou une personne habilitée).

Le système biométrique multimodal que nous avons proposé nécessite une authentification du personnel, caractérisée par un contrôle d'identité automatique et individuel à partir de trois modalités biométriques de l'usager. Ce contrôle fait appel à l'acquisition d'une image du visage en deux dimensions, et deux acquisitions de deux modalités du doigt, à savoir les empreintes digitales et le réseau vasculaire. Afin de répondre aux recommandations portant sur la protection des données personnelles et d'anticiper de futures normes, les données biométriques nécessaires à l'authentification d'un agent habilité sont stockées sur une carte sans contact nominative. Chaque personne ayant accès à la zone de stockage possède donc une carte RFID, laquelle contient ses caractéristiques biométriques de référence des trois modalités acquises par le système. Ce type de support n'étant doté que d'une faible quantité de mémoire utilisateur, la minimisation des données biométriques est un aspect que nous avons privilégié dans ces travaux.
Ce système présente deux phases principales pour chaque utilisateur. Tout d'abord une phase d'enrôlement permet d'obtenir un ensemble de données biométriques de référence, après l'acquisition et le traitement de chaque modalité. Ce sont ces données qui sont alors stockées sur la carte individuelle de l'usager. Puis, à chaque demande d'ouverture de la zone de stockage, l'utilisateur présente sa carte au système. C'est la phase d'authentification. Après avoir validé l'identifiant de la carte et le motif de la requête (gestion de missions, horaires, habilitation de la carte, etc), le système se charge de vérifier que le porteur de la carte en est bien le propriétaire. Les trois modalités sont donc mesurées et comparées aux données stockées sur la carte sans contact. Chaque modalité donne alors lieu à une décision (personne autorisée ou refusée) et l'authentification est basée sur une fusion de ces décisions au travers d'un vote à la majorité.

Les traitements des trois modalités suivent le même protocole général, traditionnellement utilisé dans la littérature pour la classification de données biométriques. Ce protocole met en œuvre des phases de prétraitements, d'extraction et de sélection de caractéristiques, ainsi qu'une phase de classification, suivie d'une potentielle réduction de données du modèle biométrique. Divers algorithmes ont été expérimentés et comparés afin d'adapter et optimiser les traitements à chaque type de données fournies par ces modalités. Les méthodes utilisées sont donc entre autres basées sur des techniques connues et matures de la littérature, fournies par des bibliothèques libres dédiées au traitement d'images (OpenCV, Sikit-Learn, etc). Nos contributions portent donc, non pas sur le développement de nouvelles méthodes, mais surtout sur les diverses possibilités d'ajustement de ces méthodes afin d'effectuer une authentification biométrique en temps réel, sur une plateforme à faible capacités de calculs (Rapsberry), à partir d'une faible quantité de données biométriques (2,6 Ko maximum par modalité, la mémoire EEPROM de la carte sans contact étant de 8 Ko).

Nous avons développé des méthodes d'authentification du visage suivant deux approches : l'une basée sur des algorithmes de type Machine Learning classique, et l'autre, à des fins de comparaison, issue d'une approche de type Deep Learning. La première approche fait appel aux traditionnelles étapes de prétraitements et d'extractions de paramètres utilisés dans la classification d'images. Deux filtres ont été expérimentés en guise de prétraitements, à savoir le filtre de Sobel et de TanTriggs, afin de réduire l'influence de variations lumineuses et d'autres perturbations. L'extraction de caractéristiques est effectuée par l'ACP grâce à laquelle un espace de visages (stocké sur l'unité de calculs) permet de représenter, sous une forme optimale, les caractéristiques issues d'une image de visage. Une réduction de données est réalisée suivant la méthode des Eigenfaces où une sélection des composantes les plus importantes de cet espace de visages permet de réduire la dimension de représentation des données. Nous avons comparé, pour chaque espace issu de l'ACP selon le filtrage de l'image, trois seuils définissant une quantité de données de plus en plus restreinte (de 299 à 7 valeurs propres, soit pour une image un vecteur de 299 à seulement 7 caractéristiques). Finalement, deux algorithmes de classification ont été comparés (SVM et RF). Une fois compressés, ce sont les modèles issus de ces classifieurs individuels qui sont stockés sur la carte sans contact, après une réduction de la précision de leurs données. La seconde approche repose sur une implémentation de CNN légers (architecture constituée de peu de paramètres) au travers du transfert de connaissances. Ces CNN intègrent toute la chaîne de traitements, du filtrage à la classification. Par conséquent, seul le paramétrage de l'apprentissage par transfert de connaissances a été l'objet de nos expérimentation (quantité d'images d'apprentissage, dimension de ces images, hyperparamètres de l'apprentissage).
Les deux modalités du doigt présentent des motifs aux structures similaires (lignes courbes composées de bifurcations et de croisement de lignes). Par conséquent, pour les empreintes et les vaisseaux, les traitements mis en place sont composés des mêmes méthodes pour lesquelles les paramètres ont été ajustés selon la modalité. De nombreux prétraitements (débruitage, filtre de Gabor, etc) permettent d'isoler le motif composé par ces lignes, sous une forme squelettisée conservant sa consistance géométrique. Cette forme facilite la détection de points d'intérêts (fins de lignes, bifurcations et croisements de lignes) qui sont ensuite décrits localement par un descripteur binaire (BRIEF). Ces descripteurs locaux sont stockés sur la carte individuelle. Ils représentent la référence à laquelle sera ensuite comparée l'ensemble des descripteurs issus de l'échantillon de la tentative d'authentification. Cette comparaison est effectuée par une mise en correspondance des descripteurs, quantifiée par plusieurs grandeurs comme la distance de Hamming (distance absolue, moyenne, écart type), le nombre de correspondances, etc. Ces grandeurs forment un vecteur d'attributs sélectionnés par SBFS et alimentant un classifieur SVM. Le modèle obtenu est stocké sur l'unité de calcul et permet d'obtenir une décision pour chacune des deux modalités (personne autorisée ou refusée) à partir des résultats de cette mise en correspondance.

Afin de répondre aux contraintes du projet, nous avons sélectionné, pour chaque modalité, les méthodes et les configurations optimales permettant d'obtenir un compromis intéressant entre la taille de stockage et les performances d'authentification. La table 7.1 présente un résumé des résultat d'authentification expérimentaux de chaque modalité seule, pour les configurations retenues, décrites dans les chapitres 4 et 5. La table 7.2 présente les résultats d'authentification expérimentaux de la fusion de décision de ces trois modalités selon un vote à la majorité et un vote à l'unanimité, à partir d'une base de données composée d'un sous ensemble des bases utilisées pour l'étude des modalités seules.

| **Modalité** | **Configuration sélectionnée** | **Se** | **Sp** | **Pr** | **Ju** | $T_{max}$ |
|---|---|---|---|---|---|---|
| **Visage** | TanTriggs / $\beta = 20$ / ACP 50 / SVM | 81,6 | 91,6 | 90,7 | 86,6 | 2,6 |
| **Empreintes** | BRIEF-16 / $\xi_C = 80$ / SVM | 77,9 | 98,9 | 98,6 | 88,4 | 2,5 |
| **Vaisseaux** | BRIEF-32 / $\xi_C = 40$ / SVM | 81,3 | 97,3 | 96,8 | 89,3 | 2,5 |

TABLE 7.1 – Résumé des performances (en pourcent) des configurations retenues de chaque modalité indépendante, obtenues expérimentalement sur l'ensemble des bases de données de chaque modalité et la taille de stockage maximale ($T_{max}$) des données biométriques (en kilooctets)

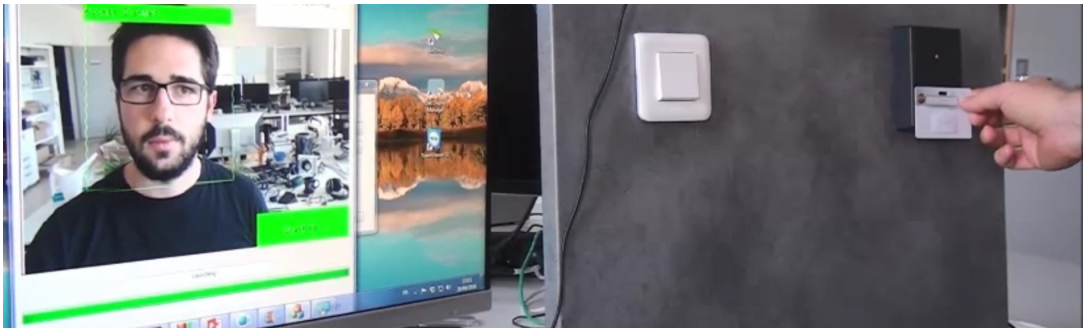
| **Type de fusion** | **Scénario** | **Se** | **Sp** | **Pr** | **Ju** | $T_{max}$ |
|---|---|---|---|---|---|---|
| **Vote à la majorité** | Intrusion | 95,5 | 100 | 100 | 97,6 | 7,6 |
| | Usurpation du visage | 95,2 | 96,9 | 96,8 | 96,1 | |
| | Usurpation des empreintes | 95,2 | 95,1 | 86,5 | 90,2 | |
| | Usurpation des vaisseaux | 95,2 | 79,6 | 82,4 | 87,4 | |
| **Vote à l'unanimité** | Intrusion | 52,8 | 100 | 100 | 76,4 | 7,6 |
| | Usurpation du visage | 52,8 | 100 | 100 | 76,4 | |
| | Usurpation des empreintes | 52,8 | 100 | 100 | 76,4 | |
| | Usurpation des vaisseaux | 52,8 | 100 | 100 | 76,4 | |

TABLE 7.2 – Résumé des performances d'authentification biométrique (en pourcent) par fusion de décision avec un vote à la majorité et un vote à l'unanimité, obtenues expérimentalement sur un sous ensemble des bases de données de chaque modalité et la taille de stockage maximale ($T_{max}$) des données biométriques (en kilooctets)

Les performances d'authentification des modalités individuelles présentent une bonne aptitude à refuser les intrus (spécificité supérieure à 90%) tandis que les personnes habilitées seront plus souvent refusées (faux négatifs : 80% de sensibilité en moyenne). La fusion de décision par le vote à la majorité permet d'améliorer largement ces performances. Dans le cas d'une simple intrusion, le système refuse systématiquement les personnes non autorisées (aucun faux positifs) et les personnes habilitées n'ont désormais que 4,5% de probabilité d'être considérées comme un faux négatif. Dans le cas de l'usurpation d'une des trois modalités, le vote à la majorité ne permet pas de conserver une restrictivité totale. De plus, dans le cas de l'usurpation des vaisseaux sanguins, le système atteint 20% de faux positifs. En revanche, le vote à l'unanimité permet de maintenir un très haut niveau de sécurité (aucun faux positifs) au détriment de l'authentification des personnes autorisées (atteignant 50% de refus).

Au travers des travaux présentés dans cette thèse, nous avons montré qu'il est possible de stocker les données biométriques de trois modalités individuelles sur une carte sans contact de type RFID. Chacune de ces données, que ce soit un classifieur pour le visage ou un ensemble de descripteurs pour le doigt, nécessite un espace de stockage inférieur à 2,6 Ko. Pour le visage, les performances brutes d'authentification ont montrées que, suivant l'espace de stockage disponible et les contraintes de l'application, il est possible d'ajuster les performances et la taille des modèles en jouant sur la contribution du filtrage des images d'entrées et la quantité de données conservées dans l'espace de visages crée par l'ACP. Pour les modalités du doigt, les performances et la taille de stockage des données biométriques peuvent être ajustées en fixant un nombre maximum de points d'intérêts traités dans une image et la précision de description du descripteur.

Seul le traitement de la modalité du visage a bénéficié d'une intégration matérielle sur une plateforme embarquée. La mesure des temps de calculs sur cette plateforme d'implantation (caméra intelligente) nous a permis d'écarter les solutions à base de Deep Learning. La caméra intelligente, composée d'un capteur d'image traditionnel et d'une carte Raspberry, permet d'effectuer une authentification du visage à une fréquence de 17 ips, soit 58 ms par image, pour la configuration retenue (Machine Learning classique). La figure 7.1 expose une évaluation du développement algorithmique dédié au visage. Cette évaluation a été effectuée sur la plateforme de développement (PC), avec un capteur d'image standard (webcam), un lecteur de carte RFID et une carte sans contact où seul un identifiant a été stocké (le modèle biométrique répondant à cet identifiant est stocké sur la plateforme de développement. Deux cas sont évalués ici. Le premier cas, figure 7.1a illustre une personne habilitée se présentant devant le système avec sa carte personnelle. Le second cas, figure 7.1b, illustre une intrusion où une personne aux caractéristiques similaires (brun, lunettes, barbe) se présente devant le système avec une carte volée, appartenant à l'individu précédent. On observe que l'intrusion est bien détectée (encadré rouge autour du visage de l'intrus).



(a) Personne autorisée avec sa propre carte (visage seulement)

(b) Personne aux caractéristiques similaires avec une carte volée - intrusion (visage seulement)

FIGURE 7.1 – Test de l'authentification de visages en implantation réelle

L'authentification de visage est soumise à de nombreuses variations entre les images d'apprentissage et les images d'acquisition, telles que des variations d'éclairage, ou encore des variations d'orientation et d'expression, etc. Les variations intrinsèques au visage du sujet (comportement de l'individu) peuvent être atténuées en imposant une contrainte de position de la tête pour l'acquisition. L'utilisation d'un "patron", dessiné sur l'interface IHM avec un retour de la vidéo (principe des cabines photographiques), est une piste intéressante pouvant aider au positionnement du visage. Bien que potentiellement atténuées par ces contraintes et la maîtrise de l'éclairage, l'authentification du visage à partir d'une unique image n'est pas judicieuse. Il parait crucial en effet d'effectuer cette authentification à partir d'une série de $N$ acquisitions, dans une certaine fenêtre temporelle, afin de réduire l'influence des variations résiduelles et de stabiliser la décision finale. Cette décision finale est alors représentée par un vote à la majorité des $N$ prédictions du classifieur sur cette fenêtre temporelle, réduisant ainsi la probabilité d'un faux rejet. De plus, couplée à l'apprentissage rapide de notre implémentation (de l'ordre de la milliseconde), une telle configuration permet d'effectuer une mise à jour régulière et transparente du modèle SVM avec les $M$ images validées parmi les $N$ acquisitions, afin de pallier aux éventuelles variations temporelles pouvant avoir une influence sur l'aspect physique du visage (vieillissement, cheveux, barbe, etc).
Nous n'avons pas retenu l'implantation des CNN pour l'authentification de visage, du fait de la vitesse de calculs bien trop faible sur notre plateforme de déploiement. Cependant, l'utilisation de ces CNN avec une clé neuronale (VPU) optimisée pour ce type de calculs, reliée à une Raspberry, permet d'augmenter considérablement la vitesse de traitements des CNN tout en ne nécessitant qu'une consommation d'1W. Il serait donc intéressant d'effectuer, à titre de comparaison, une implantation sur cette plateforme mettant en œuvre les CNN MobileNet comme simple extracteurs de caractéristiques avec un classifieur SVM individuel.

Cette authentification de visage est effectuée en deux dimensions. Par conséquent, cette modalité peut être facilement être usurpée par une simple photographie du propriétaire de la carte sans contact. Diverses solutions se présentent alors pour augmenter la robustesse à ce type d'attaque sans changer de capteur d'images. Tout d'abord, il est possible de remonter à une information en trois dimensions, à partir de plusieurs images d'un visage exposé sous divers angles d'éclairage. Avec ces images, la technique nommée *"shape from shading"* [284, 285] permet de reconstruire un modèle en trois dimensions du visage et ainsi détecter si le visage mesuré est une simple photo. De plus, l'information en trois dimensions permet d'accroitre les performances d'authentification de visage (associé à des algorithmes appropriés à ce type d'informations). Luo et.al. ont montré récemment [286] qu'il était possible de remonter à cette information en trois dimensions à partir d'une unique image. Afin de réduire les risques d'attaques avec une photo ou même un masque, il est possible d'effectuer une mesure de la fréquence cardiaque sans contact (photopléthysmographie). Déjà exploitée dans notre laboratoire [37, 287, 288], la photopléthysmographie est une technique non invasive permettant de détecter, avec une simple caméra, les variations du volume sanguin dans les capillaires sous cutanés. Cependant, elle nécessite une durée d'acquisition supérieure, la stabilité de la mesure nécessitant plusieurs dizaines de battements cardiaques. Ces deux traitements supplémentaires présentent un apport significatif à notre mesure du visage et pourront faire l'objet d'une intégration dans de futurs travaux.

D'un point de vue purement algorithmique, la robustesse de l'authentification par les empreintes digitales peut être améliorée. Comme précisé dans le chapitre 5, cette amélioration réside dans une vérification complémentaire de l'authentification par minutie. De futurs travaux pourront en effet intégrer une implémentation de l'indice de Poincaré afin d'inclure aux données biométriques une information concernant les caractéristiques de niveau 1 des empreintes, à savoir le type de singularité (aspect général de l'empreinte). Du fait de leur forme linéaire, les motifs des empreintes et des vaisseaux peuvent être traitées avec un détecteur de structures curvilinéaires. Déjà exploité

dans notre laboratoire [210], ce détecteur permet de détecter les structures linéaires dans l'image et d'extraire de nouveaux descripteurs (*"shape context"*) fournissant des informations non pas sur des points d'intérêts, mais sur la structure géométrique des lignes. Les descripteurs résultant sont en effet composés d'informations sur l'orientation des lignes, leur taille, et sur les structures présentes dans un certain voisinage. Il serait donc intéressant de comparer ces descripteurs aux développements présentés dans les travaux de cette thèse afin d'estimer la pertinence des données géométriques dans la mise en correspondance des empreintes et des vaisseaux.
D'un point de vue matériel, la partie acquisition des modalités du doigt peut également être améliorée. Le prototype de scanner rotatif proposé présente l'avantage de mesurer les empreintes digitales et les vaisseaux vasculaires sur le tour du doigt, rendant alors plus complexe l'usurpation de ces modalités et réduisant les potentielles variations de positions du doigt (translation et rotation) que l'on peut trouver avec un système d'acquisition sans contact. Cependant, le développement actuel du système ne permet pas d'effectuer une acquisition correcte des vaisseaux sanguins, et les empreintes digitales, bien que nettes, ne sont pas suffisamment contrastées. Ce prototype mérite donc de plus amples expérimentations et améliorations afin d'obtenir des acquisitions reproductibles et optimales. Nous avons vu que le maintien de la température du capteur en dessous de 0°C augmente sa sensibilité et réduit la quantité de charges accumulées par le capteur lors de l'intégration. L'utilisation d'un système de refroidissement peut donc être envisagé. Cependant, au vu de l'encombrement d'un tel système et de la liberté de mouvement nécessaire à notre prototype, son intégration est une tâche complexe.

Ces travaux de thèse ont permis de définir plusieurs chaînes algorithmiques dont la robustesse a été évaluée tout en respectant des contraintes matérielles fortes. Cette approche originale par rapport aux travaux de l'état de l'art a permis de réaliser une comparaison entre des méthodes basées sur une approche de type Machine Learning classique et une approche Deep Learning, sans se limiter aux performances brutes d'authentification, mais dans l'objectif d'une utilisation réelle dans le milieu industriel. Au travers de preuves de concept algorithmiques et de nombreuses études, cette thèse doit permettre de guider la conception d'un système de sécurité biométrique multimodal au regard de contraintes applicatives. Une preuve de concept a été réalisée d'une part avec l'implémentation matérielle des développements proposés au sein d'une caméra intelligente, et d'autre part avec la conception d'un prototype d'acquisition des empreintes digitales et des vaisseaux sanguins. Le déploiement d'un tel système de sécurité en industrie requiert naturellement de plus amples développements. Afin de pleinement garantir la protection de la vie privée des utilisateurs, les données biométriques stockées sur la carte sans contact doivent être chiffrées, au même titre que toute communication entre les caméras intelligentes et le système ou entre le système et un serveur. L'interface IHM doit être développée afin de faciliter les interactions avec les usagers.

Ces travaux ont donné lieu à des valorisations scientifiques au travers de publications et des conférences internationales. Énumérées à la page suivante, ces valorisations comprennent également un article pour une revue, en cours de rédaction, décrivant une synthèse de ces travaux.

# Valorisations scientifiques

**Conférences internationales :**

1. [289] Bonazza, P., Mitéran, J., Ginhac, D., and Dubois, J., **Comparative study of deep learning and classical methods : smart camera implementation for face authentication**, *In Counterterrorism, Crime Fighting, Forensics, and Surveillance Technologies II (2018), International Society for Optics and Photonics* - **Best student paper**
2. [290] Bonazza, P., Mitéran, J., Ginhac, D., and Dubois, J., **Machine Learning VS Transfer Learning Smart Camera Implementation for Face Authentication**, *In Proceedings of the 12th International Conference on Distributed Smart Cameras (2018), ACM*

**Conférence nationale :**

3. [218] Bonazza, P., Mitéran, J., Ginhac, D., and Dubois, J., **Optimisation conjointe de la taille de stockage et des performances de modèles de classification pour l'authentification de visages**, *In GRETSI (2017)*

**Wrokshop international :**

4. [291] Bonazza, P., Mitéran, J., Heyrman, B., Ginhac, D., Thivent, V., and Dubois, J., **An affordable contactless security system access for restricted area**, *In 5th Workshop on Architecture of Smart Cameras WASC (2016)*
5. [292] Dubois, J., Moinet, A. Bobbia, S., Marroquin, R., Heyrman, B., Bonazza, P., et.al., **WiseEye : A Platform to Manage and Experiment on Smart Camera Networks**, *In 5th Workshop on Architecture of Smart Cameras WASC (2016)*

**Articles :**

6. **Article de synthèse pour une revue, en cours de rédaction.**
   Bonazza, P., Mitéran, J., Ginhac, D., and Dubois, J., **Multimodal biometric authentication in hardware resources and storage constrained system**

# Bibliographie


[1] Assystem, I. F. **Nuc-track : Une solution intelligente, efficace et innovante pour le suivi et la traçabilité de sources radioactives**. Tech. rep., 2014.

[2] IRSN. **Le transport de matières radioactives**. Tech. rep., 2007.

[3] ASN. **Le transport de substances radioactives - fiche d'information du public n°8**. Tech. rep., 2014.

[4] ASN. **Rapport de l'asn sur l'état de la sûreté nucléaire et de la radioprotection en france en 2017 - chapitre 11**. Tech. rep., 2017.

[5] IRSN. **Sûreté des transports de substances radioactives à usage civil sur le territoire français - rapport de mission 2016**. Tech. rep., 2016.

[6] Assystem, I. F. **Nuc-track presentation**. Tech. rep., 2014.

[7] Weinland, D., Ronfard, R., and Boyer, E. **A survey of vision-based methods for action representation, segmentation and recognition**. *Computer vision and image understanding 115*, 2 (2011), 224–241.

[8] Herath, S., Harandi, M., and Porikli, F. **Going deeper into action recognition : A survey**. *Image and vision computing 60* (2017), 4–21.

[9] Choras, M. **Image feature extraction methods for ear biometrics–a survey**. In *Computer Information Systems and Industrial Management Applications, 2007. CISIM'07. 6th International Conference on* (2007), IEEE, pp. 261–265.

[10] Choraś, M. **The lip as a biometric**. *Pattern Analysis and Applications 13*, 1 (2010), 105–112.

[11] Garg, S., Kumar, A., and Hanmandlu, M. **Biometric authentication using finger nail surface**. In *Intelligent Systems Design and Applications (ISDA), 2012 12th International Conference on* (2012), IEEE, pp. 497–502.

[12] Wu, S., Lin, W., and Xie, S. **Skin heat transfer model of facial thermograms and its application in face recognition**. *Pattern Recognition 41*, 8 (2008), 2718–2729.

[13] Plataniotis, K. N., Hatzinakos, D., and Lee, J. K. **Ecg biometric recognition without fiducial detection**. In *Biometric Consortium Conference, 2006 Biometrics Symposium : Special Session on Research at the* (2006), IEEE, pp. 1–6.

[14] Jain, A. K., Ross, A., and Prabhakar, S. **An introduction to biometric recognition**. *IEEE Transactions on circuits and systems for video technology 14*, 1 (2004), 4–20.

[15] Sagayam, K. M., Edwin, J. F. J., Christopher, J. S., Reddy, G. V., Bestak, R., and Hun, L. C. **Survey on the classification of intelligence-based biometric techniques**. In *Biologically Rationalized Computing Techniques For Image Processing Applications*. Springer, 2018, pp. 111–132.

[16] Jafri, R., and Arabnia, H. R. **A survey of face recognition techniques.** *Jips 5*, 2 (2009), 41–68.

[17] Sharif, M., Naz, F., Yasmin, M., Shahid, M. A., and Rehman, A. **Face recognition : A survey.** *Journal of Engineering Science & Technology Review 10*, 2 (2017).

[18] Kakadiaris, I. A., Passalis, G., Toderici, G., Efraty, E., Perakis, P., Chu, D., Shah, S., and Theoharis, T. **Face Recognition Using 3D Images**. Springer London, 2011, pp. 429–459.

[19] Rajanna, U., Erol, A., and Bebis, G. **A comparative study on feature extraction for fingerprint classification and performance improvements using rank-level fusion**. *Pattern Analysis and Applications 13*, 3 (2010), 263–272.

[20] Davis, D., Higgins, P., Kormarinski, P., Marques, J., Orlans, N., and Wayman, J. **State of the art biometrics excellence roadmap**. *MITRE Corporation : Bedford, MA, USA 1* (2008).

[21] Vlachos, M., and Dermatas, E. **Finger vein segmentation in infrared images using supervised and unsupervised clustering algorithms**. *Pattern recognition and image analysis 23*, 2 (2013), 328–334.

[22] Sanchez-Reillo, R., Sanchez-Avila, C., and Gonzalez-Marcos, A. **Biometric identification through hand geometry measurements**. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 10 (2000), 1168–1171.

[23] Bhuiyan, A., Nath, B., and Ramamohanarao, K. **Detection and classification of bifurcation and branch points on retinal vascular network**. In *International Conference on Digital Image Computing Techniques and Applications, DICTA 2012* (2012).

[24] Daugman, J. **Chapter 25 - how iris recognition works**. In *The Essential Guide to Image Processing*, A. Bovik, Ed. Academic Press, Boston, 2009, pp. 715 – 739.

[25] Pflug, A., and Busch, C. **Ear biometrics : a survey of detection, feature extraction and recognition methods**. *IET Biometrics 1*, 2 (2012).

[26] Yampolskiy, R. V., and Govindaraju, V. **Behavioural biometrics : a survey and classification**. *International Journal of Biometrics 1*, 1 (2008), 81–113.

[27] Pato, J., and Millett, L. **Biometric recognition : Challenges and opportunities, whither biometrics committee, national research council of the nsa**, 2010.

[28] Mailah, M., and Lim, B. H. **Biometric signature verification using pen position, time, velocity and pressure parameters**. *University Technology. Malaysia* (2008).

[29] Monrose, F., and Rubin, A. D. **Keystroke dynamics as a biometric for authentication**. *Future Generation computer systems 16*, 4 (2000), 351–359.

[30] Shah, H. N. M., Ab Rashid, M. Z., Abdollah, M. F., Kamarudin, M. N., Chow, K. L., and Kamis, Z. **Biometric voice recognition in security system**. *Indian Journal of Science and Technology 7*, 2 (2014), 104–112.

[31] Soram, R., Khomdram, M., and others. **Biometric dna and ecdlp-based personal authentication system : a superior posse of security**. *IJCSNS 10*, 1 (2010), 1.

[32] Pal, A., and Singh, Y. N. **Ecg biometric recognition**. In *Mathematics and Computing* (Singapore, 2018), D. Ghosh, D. Giri, R. N. Mohapatra, E. Savas, K. Sakurai, and L. P. Singh, Eds., Springer Singapore, pp. 61–73.

[33] Sonkamble, S., Thool, D. R., and Sonkamble, B. **Survey of biometric recognition systems and their applications.** *Journal of Theoretical & Applied Information Technology 11* (2010).

[34] Prabhakar, S., Pankanti, S., and Jain, A. K. **Biometric recognition : Security and privacy concerns**. *IEEE security & privacy*, 2 (2003), 33–42.

[35] Bhatia, R. **Biometrics and face recognition techniques**. *International Journal of Advanced Research in Computer Science and Software Engineering 3*, 5 (2013).

**[36]** Liu, Y., Jourabloo, A., and Liu, X. **Learning deep models for face anti-spoofing : Binary or auxiliary supervision**. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2018), pp. 389–398.

**[37]** Bobbia, S., Macwan, R., Benezeth, Y., Mansouri, A., and Dubois, J. **Unsupervised skin tissue segmentation for remote photoplethysmography**. *Pattern Recognition Letters* (2017).

**[38]** De Luca, A., Hang, A., Von Zezschwitz, E., and Hussmann, H. **I feel like i'm taking selfies all day ! : towards understanding biometric authentication on smartphones**. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems* (2015), ACM, pp. 1411–1414.

**[39]** Clarke, N. L., and Furnell, S. M. **Authentication of users on mobile telephones–a survey of attitudes and practices**. *Computers & Security 24*, 7 (2005), 519–527.

**[40]** Meng, W., Wong, D. S., Furnell, S., and Zhou, J. **Surveying the development of biometric user authentication on mobile phones**. *IEEE Communications Surveys & Tutorials 17*, 3 (2015), 1268–1293.

**[41]** Brant, C., Steven Martin, R., and Frederick J., Z. **Receiving fingerprints through touch screen of ce device**, 2017. Sony Corp, Granted patent number US9646307B2.

**[42]** Yi, H., and Bo, P. **Under-lcd screen optical sensor module for on-screen fingerprint sensing**, 2018. Shenzhen Goodix Tech Co Ltd, Application patent number US20180005005A1.

**[43]** Yi, H., and Bo, P. **Optical sensing performance of under-screen optical sensor module for on-screen fingerprint sensing**, 2018. Shenzhen Goodix Tech Co Ltd, Application patent number US20180129798A1.

**[44]** Oganesian, V. **Under screen sensor assembly**, 2018. Optiz Inc, Granted patent number US9996725B2.

**[45]** Benny, P., and Zafrir, M. **Projectors of structured light**, 2016. Apple Inc., Pending patent number US15265877.

**[46]** Kwang-hyuk, B., Tae-Chan, K., Seong-Yeong, J., and Shung-Han, C. **Electronic device and eye region detection method in electronic device**, 2016. Samsung Electronics Co Ltd , Granted patent number US9418306B2.

**[47]** Hintze, D., Findling, R. D., Scholz, S., and Mayrhofer, R. **Mobile device usage characteristics : The effect of context and form factor on locked and unlocked usage**. In *Proceedings of the 12th International Conference on Advances in Mobile Computing and Multimedia* (2014), ACM, pp. 105–114.

**[48]** Bhagavatula, R., Ur, B., Iacovino, K., Kywe, S. M., Cranor, L. F., and Savvides, M. **Biometric authentication on iphone and android : Usability, perceptions, and influences on adoption**.

**[49]** **Facestation 2 - smart face recognition terminal**. Tech. rep. https://www.supremainc.com/sites/all/themes/suprema/images/products/facestation2/[AHL-FST2-180110-06-EN].pdf.

**[50]** **Biostation 2 - ultra performance ip fingerprint terminal**. Tech. rep. https://cdn2.hubspot.net/hubfs/1783147/Brochures%20(ACM,%20Suprema,%20Cabinet)/biostation2_brochure_EN-ETS-20150420.pdf.

**[51]** **Hybrid finger identification**. Tech. rep. https://uk.nec.com/en_GB/global/solutions/safety/Products/hybrid_finger.html.

**[52]** **Morpho 3d face reader - fast, convenient, secure facial recognition**. Tech. rep. https://www.morpho.com/sites/morpho/files/morpho_3d_face_reader_en_0.pdf.

[53] **Safran morphowave tower - access control with a wave of the hand**. Tech. rep. https://www.morpho.com/sites/morpho/files/morphowave-tower-032016-en.pdf.

[54] **Safran morphosmart finger vp - modile biométrique hybride réseau veineux et empreinte digitale**. Tech. rep. https://www.morpho.com/sites/morpho/files/morphosmart-finger-vp-oem-series_-012016-fr.pdf.

[55] **Lecteur biométrique veinaccess - reconnaissance par les veines de la main**. Tech. rep. http://www.zalix.fr/images/docs/VeinAccess/Docs/Doc%20VeinAccess.pdf.

[56] **Hybrid biometric platform : 1 integration, 1 server – 4 leading forms of biometric technology**. Tech. rep. http://www.m2sys.com/wp-content/uploads/pdf/Hybrid-Biometric-Platform.pdf.

[57] **Think excellence, choose multimodality**. Tech. rep. https://www.morpho.com/sites/morpho/files/multimodality_en-5_0.pdf.

[58] Lumini, A., and Nanni, L. **Overview of the combination of biometric matchers**. *Information Fusion 33* (2017), 71–85.

[59] Garg, S. N., Vig, R., and Gupta, S. **A survey on different levels of fusion in multimodal biometrics**. *Indian Journal of Science and Technology 10*, 44 (2017).

[60] Ross, A., and Jain, A. **Information fusion in biometrics**. *Pattern recognition letters 24*, 13 (2003), 2115–2125.

[61] Guesmi, H. **Identification de personnes par fusion de différentes modalités biométriques**. PhD thesis, Télécom Bretagne ; Université de Rennes 1, 2014.

[62] Morizet, N. **Reconnaissance Biométrique par Fusion Multimodale du Visage et de l'Iris**. Theses, Télécom ParisTech, 2009.

[63] Iyengar, S., Prasad, L., and Min, H. **Advances in distributed sensor technology**, 1995.

[64] Woods, K., Kegelmeyer, W. P., and Bowyer, K. **Combination of multiple classifiers using local accuracy estimates**. *IEEE transactions on pattern analysis and machine intelligence 19*, 4 (1997), 405–410.

[65] Huang, Y. S., and Suen, C. Y. **A method of combining multiple experts for the recognition of unconstrained handwritten numerals**. *IEEE transactions on pattern analysis and machine intelligence 17*, 1 (1995), 90–94.

[66] Kittler, J., Hatef, M., Duin, R. P., and Matas, J. **On combining classifiers**. *IEEE transactions on pattern analysis and machine intelligence 20*, 3 (1998), 226–239.

[67] Lupu, E., and Pop, P. G. **Multimodal biometric systems overview**.

[68] Ross, A., Jain, A., and Reisman, J. **A hybrid fingerprint matcher, pattern recognition vol. 36, 2003 pp. 1661-1673**.

[69] Ho, T. K., Hull, J. J., and Srihari, S. N. **Decision combination in multiple classifier systems**. *IEEE transactions on pattern analysis and machine intelligence 16*, 1 (1994), 66–75.

[70] Lam, L., and Suen, S. **Application of majority voting to pattern recognition : an analysis of its behavior and performance**. *IEEE Transactions on Systems, Man, and Cybernetics-Part A : Systems and Humans 27*, 5 (1997), 553–568.

[71] Lam, L., and Suen, C. Y. **Optimal combinations of pattern classifiers**. *Pattern Recognition Letters 16*, 9 (1995), 945–954.

[72] Xu, L., Krzyzak, A., and Suen, C. Y. **Methods of combining multiple classifiers and their applications to handwriting recognition**. *IEEE transactions on systems, man, and cybernetics 22*, 3 (1992), 418–435.

**[73]** Krishneswari, K., and Arumugam, S. **Multimodal biometrics using feature fusion**. *Journal of Computer Science 8*, 3 (2012), 431.

**[74]** Lahane, P., and Ganorkar, S. **Fusion of iris & fingerprint biometric for security purpose**. *International Journal of Scientific & Engineering Research 3*, 8 (2012), 1–5.

**[75]** Abdolahi, M., Mohamadi, M., and Jafari, M. **Multimodal biometric system fusion using fingerprint and iris with fuzzy logic**. *International Journal of Soft Computing and Engineering 2*, 6 (2013), 504–510.

**[76]** Cui, F., and Yang, G. **Score level fusion of fingerprint and finger vein recognition**. *Journal of Computational information systems 7*, 16 (2011), 5723–5731.

**[77]** Choras, R. S. **Image feature extraction techniques and their applications for cbir and biometrics systems**. *International journal of biology and biomedical engineering 1*, 1 (2007), 6–16.

**[78]** Chihaoui, M., Elkefi, A., Bellil, W., and Ben Amar, C. **A survey of 2d face recognition techniques**. *Computers 5*, 4 (2016), 21.

**[79]** Pearson, K. **Liii. on lines and planes of closest fit to systems of points in space**. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science 2*, 11 (1901), 559–572.

**[80]** Geladi, P., Isaksson, H., Lindqvist, L., Wold, S., and Esbensen, K. **Principal component analysis of multivariate images**. *Chemometrics and Intelligent Laboratory Systems 5*, 3 (1989), 209–220.

**[81]** Wold, S., Esbensen, K., and Geladi, P. **Principal component analysis**. *Chemometrics and intelligent laboratory systems 2*, 1-3 (1987), 37–52.

**[82]** Ans, B. **Adaptive neural architectures**. *Detection of primitives. Proceedings of COGNITIVA'85* (1985), 593–597.

**[83]** Comon, P. **Independent component analysis, a new concept ?** *Signal processing 36*, 3 (1994), 287–314.

**[84]** Riffenburgh, R. H. **Linear discriminant analysis**. PhD thesis, Virginia Polytechnic Institute, 1957.

**[85]** Shawe-Taylor, J., Cristianini, N., and others. **Kernel methods for pattern analysis**. Cambridge university press, 2004.

**[86]** Hoffmann, H. **Kernel pca for novelty detection**. *Pattern recognition 40*, 3 (2007), 863–874.

**[87]** Bach, F. R., and Jordan, M. I. **Kernel independent component analysis**. *Journal of machine learning research 3*, Jul (2002), 1–48.

**[88]** Zhang, T., Fang, B., Tang, Y. Y., Shang, Z., and Xu, B. **Generalized discriminant analysis : A matrix exponential approach**. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics) 40*, 1 (2010), 186–197.

**[89]** Brunelli, R., and Poggio, T. **Face recognition : Features versus templates**. *IEEE transactions on pattern analysis and machine intelligence 15*, 10 (1993), 1042–1052.

**[90]** Viola, P., and Jones, M. J. **Robust real-time face detection**. *International journal of computer vision 57*, 2 (2004), 137–154.

**[91]** Lowe, D. G. **Distinctive image features from scale-invariant keypoints**. *International journal of computer vision 60*, 2 (2004), 91–110.

**[92]** Pietikäinen, M., Hadid, A., Zhao, G., and Ahonen, T. **Local binary patterns for still images**. In *Computer vision using local binary patterns*. Springer, 2011, pp. 13–47.

**[93]** Heikkilä, M., Pietikäinen, M., and Schmid, C. **Description of interest regions with local binary patterns**. *Pattern recognition 42*, 3 (2009), 425–436.

**[94]** Lades, M., Vorbruggen, J. C., Buhmann, J., Lange, J., Von Der Malsburg, C., Wurtz, R. P., and Konen, W. **Distortion invariant object recognition in the dynamic link architecture**. *IEEE Transactions on computers*, 3 (1993), 300–311.

**[95]** Lee, T. S. **Image representation using 2d gabor wavelets**. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 10 (1996), 959–971.

**[96]** Aggarwal, C. C., Hinneburg, A., and Keim, D. A. **On the surprising behavior of distance metrics in high dimensional space**. In *International conference on database theory* (2001), Springer, pp. 420–434.

**[97]** Ayodele, T. O. **Types of machine learning algorithms**. In *New advances in machine learning*. InTech, 2010.

**[98]** Peng, C.-Y. J., Lee, K. L., and Ingersoll, G. M. **An introduction to logistic regression analysis and reporting**. *The journal of educational research 96*, 1 (2002), 3–14.

**[99]** Widrow, B., and Lehr, M. A. **30 years of adaptive neural networks : perceptron, madaline, and backpropagation**. *Proceedings of the IEEE 78*, 9 (1990), 1415–1442.

**[100]** Cortes, C., and Vapnik, V. **Support-vector networks**. *Machine learning 20*, 3 (1995), 273–297.

**[101]** Jain, A. K. **Data clustering : 50 years beyond k-means**. *Pattern recognition letters 31*, 8 (2010), 651–666.

**[102]** Mucherino, A., Papajorgji, P. J., and Pardalos, P. M. **K-nearest neighbor classification**. In *Data Mining in Agriculture*. Springer, 2009, pp. 83–106.

**[103]** Quinlan, J. R. **Induction of decision trees**. *Machine learning 1*, 1 (1986), 81–106.

**[104]** Breiman, L. **Random forests**. *Machine learning 45*, 1 (2001), 5–32.

**[105]** Friedman, N., Geiger, D., and Goldszmidt, M. **Bayesian network classifiers**. *Machine learning 29*, 2-3 (1997), 131–163.

**[106]** Lippmann, R. P. **An introduction to computing with neural nets**. In *Artificial neural networks : theoretical concepts* (1988), IEEE Computer Society Press, pp. 36–54.

**[107]** Kotsiantis, S. B., Zaharakis, I., and Pintelas, P. **Supervised machine learning : A review of classification techniques**. *Emerging artificial intelligence applications in computer engineering 160* (2007), 3–24.

**[108]** Biau, G., and Scornet, E. **A random forest guided tour**. *Test 25*, 2 (2016), 197–227.

**[109]** Louppe, G. **Understanding random forests : From theory to practice**. *arXiv preprint arXiv :1407.7502* (2014).

**[110]** Rumelhart, D. E., Hinton, G. E., and Williams, R. J. **Learning representations by back-propagating errors**. *nature 323*, 6088 (1986), 533.

**[111]** Dechter, R. **Learning while searching in constraint-satisfaction problems**. University of California, Computer Science Department, Cognitive Systems Laboratory, 1986.

**[112]** LeCun, Y., Bengio, Y., and Hinton, G. **Deep learning**. *nature 521*, 7553 (2015), 436.

**[113]** Schmidhuber, J. **Deep learning in neural networks : An overview**. *Neural networks 61* (2015), 85–117.

**[114]** Montavon, G., Samek, W., and Müller, K.-R. **Methods for interpreting and understanding deep neural networks**. *Digital Signal Processing* (2017).

**[115]** Olah, C., Satyanarayan, A., Johnson, I., Carter, S., Schubert, L., Ye, K., and Mordvintsev, A. **The building blocks of interpretability**. *Distill* (2018). https ://distill.pub/2018/building-blocks.

**[116]** Olah, C., Mordvintsev, A., and Schubert, L. **Feature visualization**. *Distill* (2017). https ://distill.pub/2017/feature-visualization.

**[117]** Turk, M., and Pentland, A. **Eigenfaces for recognition**. *Journal of cognitive neuroscience 3*, 1 (1991), 71–86.

**[118]** Canziani, A., Paszke, A., and Culurciello, E. **An analysis of deep neural network models for practical applications**. *arXiv preprint arXiv :1605.07678* (2016).

**[119]** Ma, N., Zhang, X., Zheng, H.-T., and Sun, J. **Shufflenet v2 : Practical guidelines for efficient cnn architecture design**. *arXiv preprint arXiv :1807.11164* (2018).

**[120]** Yao, S., Zhao, Y., Shao, H., Liu, S., Liu, D., Su, L., and Abdelzaher, T. **Fastdeepiot : Towards understanding and optimizing neural network execution time on mobile and embedded devices**. In *Proceedings of the 16th ACM Conference on Embedded Networked Sensor Systems* (2018), ACM, pp. 278–291.

**[121]** Santos, A. G., de Souza, C. O., Zanchettin, C., Macedo, D., Oliveira, A. L., and Ludermir, T. **Reducing squeezenet storage size with depthwise separable convolutions**. In *2018 International Joint Conference on Neural Networks (IJCNN)* (2018), IEEE, pp. 1–6.

**[122]** Elordi, U., Unzueta, L., Arganda-Carreras, I., and Otaegui, O. **How can deep neural networks be generated efficiently for devices with limited resources ?** In *International Conference on Articulated Motion and Deformable Objects* (2018), Springer, pp. 24–33.

**[123]** Rusu, C.-I., and Czibula, G. **Optimizing convolutional neural networks for low-resource devices**. In *2018 IEEE 14th International Conference on Intelligent Computer Communication and Processing (ICCP)* (2018), IEEE, pp. 245–252.

**[124]** Han, S., Mao, H., and Dally, W. J. **Deep compression : Compressing deep neural networks with pruning, trained quantization and huffman coding**. *arXiv preprint arXiv :1510.00149* (2015).

**[125]** Kim, Y.-D., Park, E., Yoo, S., Choi, T., Yang, L., and Shin, D. **Compression of deep convolutional neural networks for fast and low power mobile applications**. *arXiv preprint arXiv :1511.06530* (2015).

**[126]** Wu, J., Leng, C., Wang, Y., Hu, Q., and Cheng, J. **Quantized convolutional neural networks for mobile devices**. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2016), pp. 4820–4828.

**[127]** Athar, A. **An overview of datatype quantization techniques for convolutional neural networks**. *arXiv preprint arXiv :1808.07530* (2018).

**[128]** Zhao, W., Chellappa, R., Phillips, P. J., and Rosenfeld, A. **Face recognition : A literature survey**. *ACM computing surveys (CSUR) 35*, 4 (2003), 399–458.

**[129]** Kalal, Z., Mikolajczyk, K., and Matas, J. **Face-tld : Tracking-learning-detection applied to faces**. In *Image Processing (ICIP), 2010 17th IEEE International Conference on* (2010), IEEE, pp. 3789–3792.

**[130]** Kumar, N., Berg, A. C., Belhumeur, P. N., and Nayar, S. K. **Attribute and simile classifiers for face verification**. In *Computer Vision, 2009 IEEE 12th International Conference on* (2009), IEEE, pp. 365–372.

**[131]** Fu, Y., Guo, G., and Huang, T. S. **Age synthesis and estimation via faces : A survey**. *IEEE transactions on pattern analysis and machine intelligence 32*, 11 (2010), 1955–1976.

**[132]** Wang, Y., Zhang, L., Liu, Z., Hua, G., Wen, Z., Zhang, Z., and Samaras, D. **Face relighting from a single image under arbitrary unknown lighting conditions**. *IEEE Transactions on Pattern Analysis and Machine Intelligence 31*, 11 (2009), 1968–1984.

**[133]** Hatem, H., Beiji, Z., and Majeed, R. **A survey of feature base methods for human face detection**. *International Journal of Control and Automation 8*, 5 (2015), 61–78.

**[134]** Yang, M.-H., Kriegman, D. J., and Ahuja, N. **Detecting faces in images : A survey**. *IEEE Transactions on pattern analysis and machine intelligence 24*, 1 (2002), 34–58.

**[135]** Viola, P., and Jones, M. **Rapid object detection using a boosted cascade of simple features**. In *Proceedings of the Computer Society Conference on Computer Vision and Pattern Recognition* (2001), vol. 1.

**[136]** Freund, Y., Schapire, R., and Abe, N. **A short introduction to boosting**. *Journal-Japanese Society For Artificial Intelligence 14*, 771-780 (1999), 1612.

**[137]** Dalal, N., and Triggs, B. **Histograms of oriented gradients for human detection**. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on* (2005), vol. 1, IEEE, pp. 886–893.

**[138]** Bay, H., Ess, A., Tuytelaars, T., and Van Gool, L. **Speeded-up robust features (surf)**. *Computer vision and image understanding 110*, 3 (2008), 346–359.

**[139]** Freund, Y., and Schapire, R. E. **A decision-theoretic generalization of on-line learning and an application to boosting**. *Journal of computer and system sciences 55*, 1 (1997), 119–139.

**[140]** Krizhevsky, A., Sutskever, I., and Hinton, G. E. **Imagenet classification with deep convolutional neural networks**. In *Advances in neural information processing systems* (2012), pp. 1097–1105.

**[141]** Huang, G. B., Jones, M. J., and Learned-Miller, E. **Lfw results using a combined nowak plus merl recognizer**. In *Workshop on Faces in'Real-Life'Images : Detection, Alignment, and Recognition* (2008).

**[142]** Jain, V., and Learned-Miller, E. **Fddb : A benchmark for face detection in unconstrained settings**. Tech. rep., Technical Report UM-CS-2010-009, University of Massachusetts, Amherst, 2010.

**[143]** Bradski, G., and Kaehler, A. **Learning OpenCV : Computer vision with the OpenCV library**. " O'Reilly Media, Inc.", 2008.

**[144]** Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., and others. **Tensorflow : a system for large-scale machine learning.** In *OSDI* (2016), vol. 16, pp. 265–283.

**[145]** Zafeiriou, S., Zhang, C., and Zhang, Z. **A survey on face detection in the wild : past, present and future**. *Computer Vision and Image Understanding 138* (2015), 1–24.

**[146]** Yang, S., Luo, P., Loy, C. C., and Tang, X. **Faceness-net : Face detection through deep facial part responses**. *IEEE transactions on pattern analysis and machine intelligence 40*, 8 (2018), 1845–1859.

**[147]** Ranjan, R., Patel, V. M., and Chellappa, R. **Hyperface : A deep multi-task learning framework for face detection, landmark localization, pose estimation, and gender recognition**. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2017).

**[148]** Jiang, H., and Learned-Miller, E. **Face detection with the faster r-cnn**. In *Automatic Face & Gesture Recognition (FG 2017), 2017 12th IEEE International Conference on* (2017), IEEE, pp. 650–657.

**[149]** Felzenszwalb, P., McAllester, D., and Ramanan, D. **A discriminatively trained, multiscale, deformable part model**. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on* (2008), IEEE, pp. 1–8.

**[150]** Felzenszwalb, P. F., and Huttenlocher, D. P. **Pictorial structures for object recognition**. *International journal of computer vision 61*, 1 (2005), 55–79.

**[151]** Mathias, M., Benenson, R., Pedersoli, M., and Van Gool, L. **Face detection without bells and whistles**. In *European Conference on Computer Vision* (2014), Springer, pp. 720–735.

**[152]** Chen, D., Ren, S., Wei, Y., Cao, X., and Sun, J. **Joint cascade face detection and alignment**. In *European Conference on Computer Vision* (2014), Springer, pp. 109–122.

**[153]** Zhu, X., and Ramanan, D. **Face detection, pose estimation, and landmark localization in the wild**. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on* (2012), IEEE, pp. 2879–2886.

**[154]** Delac, K., Grgic, M., and Grgic, S. **Independent comparative study of pca, ica, and lda on the feret data set**. *International Journal of Imaging Systems and Technology 15*, 5 (2005), 252–260.

**[155]** Turk, M. A., and Pentland, A. P. **Face recognition using eigenfaces**. In *Computer Vision and Pattern Recognition, 1991. Proceedings CVPR'91., IEEE Computer Society Conference on* (1991), IEEE, pp. 586–591.

**[156]** Bartlett, M. S., Movellan, J. R., and Sejnowski, T. J. **Face recognition by independent component analysis**. *IEEE transactions on neural networks/a publication of the IEEE Neural Networks Council 13*, 6 (2002), 1450.

**[157]** Sharkas, M., and Elenien, M. A. **Eigenfaces vs. fisherfaces vs. ica for face recognition ; a comparative study**. In *Signal Processing, 2008. ICSP 2008. 9th International Conference on* (2008), IEEE, pp. 914–919.

**[158]** Shyam, R., and Singh, Y. N. **Evaluation of eigenfaces and fisherfaces using bray curtis dissimilarity metric**. In *Industrial and Information Systems (ICIIS), 2014 9th International Conference on* (2014), IEEE, pp. 1–6.

**[159]** Mališek, J. M., and Pavlovicová, M. O. **Support vector machines, pca and lda in face recognition**. *J. Electr. Eng 59* (2008), 203–209.

**[160]** Wiskott, L., Fellous, J.-M., Krüger, N., and Von Der Malsburg, C. **Face recognition by elastic bunch graph matching**. In *International Conference on Computer Analysis of Images and Patterns* (1997), Springer, pp. 456–463.

**[161]** Ahonen, T., Hadid, A., and Pietikainen, M. **Face description with local binary patterns : Application to face recognition**. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 12 (2006), 2037–2041.

**[162]** Tan, X., and Triggs, B. **Enhanced local texture feature sets for face recognition under difficult lighting conditions**. *IEEE transactions on image processing 19*, 6 (2010), 1635–1650.

**[163]** Déniz, O., Bueno, G., Salido, J., and De la Torre, F. **Face recognition using histograms of oriented gradients**. *Pattern Recognition Letters 32*, 12 (2011), 1598–1603.

**[164]** Samaria, F. S., and Harter, A. C. **Parameterisation of a stochastic model for human face identification**. In *Applications of Computer Vision, Proceedings of the Second IEEE Workshop on* (1994), IEEE, pp. 138–142.

**[165]** Phillips, P. J., Wechsler, H., Huang, J., and Rauss, P. J. **The feret database and evaluation procedure for face-recognition algorithms**. *Image and vision computing 16*, 5 (1998), 295–306.

[166] Schroff, F., Kalenichenko, D., and Philbin, J. **Facenet : A unified embedding for face recognition and clustering**. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (2015), pp. 815–823.

[167] Sun, Y., Wang, X., and Tang, X. **Deep learning face representation from predicting 10,000 classes**. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2014), pp. 1891–1898.

[168] Zhou, E., Cao, Z., and Yin, Q. **Naive-deep face recognition : Touching the limit of lfw benchmark or not ?** *arXiv preprint arXiv :1501.04690* (2015).

[169] Huang, G. B., Mattar, M., Berg, T., and Learned-Miller, E. **Labeled faces in the wild : A database forstudying face recognition in unconstrained environments**. In *Workshop on faces in'Real-Life'Images : detection, alignment, and recognition* (2008).

[170] Huang, G. B., and Learned-Miller, E. **Labeled faces in the wild : Updates and new reporting procedures**. *Dept. Comput. Sci., Univ. Massachusetts Amherst, Amherst, MA, USA, Tech. Rep* (2014), 14–003.

[171] Balaban, S. **Deep learning and face recognition : the state of the art**. In *Biometric and Surveillance Technology for Human and Activity Identification XII* (2015), vol. 9457, International Society for Optics and Photonics, p. 94570B.

[172] Kulwinder, S., Kiranbir, K., and Ashok, S. **Fingerprint feature extraction**. *IJCST 2*, 3 (2011), 237–241.

[173] Hong, L., Wan, Y., and Jain, A. **Fingerprint image enhancement : Algorithm and performance evaluation**. *IEEE transactions on pattern analysis and machine intelligence 20*, 8 (1998), 777–789.

[174] Yager, N., and Amin, A. **Fingerprint verification based on minutiae features : a review**. *Pattern Analysis and Applications 7*, 1 (2004), 94–113.

[175] Henry, E. R. **Classification and uses of finger prints**. HM Stationery Office, 1905.

[176] Syifaa'Ahmad, A., Hassan, R., Ibrahim, N., Ahmad, M. N., and Ramlan, R. **An investigation of matching approaches in fingerprints identification**. In *Recent Developments in Intelligent Computing, Communication and Devices*. Springer, 2017, pp. 9–15.

[177] Peralta, D., Galar, M., Triguero, I., Paternain, D., García, S., Barrenechea, E., Benítez, J. M., Bustince, H., and Herrera, F. **A survey on fingerprint minutiae-based local matching for verification and identification : Taxonomy and experimental evaluation**. *Information Sciences 315* (2015), 67–87.

[178] Liu, C., Xia, T., and Li, H. **A hierarchical hough transform for fingerprint matching**. In *Biometric Authentication*. Springer, 2004, pp. 373–379.

[179] Zhu, E., Yin, J., and Zhang, G. **Fingerprint matching based on global alignment of multiple reference minutiae**. *Pattern Recognition 38*, 10 (2005), 1685–1694.

[180] Chen, F., Huang, X., and Zhou, J. **Hierarchical minutiae matching for fingerprint and palmprint identification**. *IEEE Transactions on Image Processing 22*, 12 (2013), 4964–4971.

[181] Kumar, R. **A review of non-minutiae based fingerprint features**. *International Journal of Computer Vision and Image Processing (IJCVIP) 8*, 1 (2018), 32–58.

[182] Yang, J. **Non-minutiae based fingerprint descriptor**. In *Biometrics*. InTech, 2011.

[183] Kovacs-Vajna, Z. M. **A fingerprint verification system based on triangular matching and dynamic time warping**. *IEEE Transactions on Pattern Analysis and Machine Intelligence 22*, 11 (2000), 1266–1276.

**[184]** Nilsson, K., and Bigun, J. **Localization of corresponding points in fingerprints by complex filtering**. *Pattern Recognition Letters 24*, 13 (2003), 2135–2144.

**[185]** Wilson, C. L., Watson, C. I., and Paek, E. G. **Effect of resolution and image quality on combined optical and neural network fingerprint matching**. *Pattern Recognition 33*, 2 (2000), 317–331.

**[186]** Jin, A. T. B., Ling, D. N. C., and Song, O. T. **An efficient fingerprint verification system using integrated wavelet and fourier–mellin invariant transform**. *Image and Vision Computing 22*, 6 (2004), 503–513.

**[187]** Helfroush, S., and Ghassemian, H. **Nonminutiae-based decision-level fusion for fingerprint verification**. *EURASIP Journal on Applied Signal Processing 2007*, 1 (2007), 35–35.

**[188]** Jain, A. K., Prabhakar, S., Hong, L., and Pankanti, S. **Fingercode : a filterbank for fingerprint representation and matching**. In *Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on.* (1999), vol. 2, IEEE, pp. 187–193.

**[189]** Dai, X., Liang, J., Zhao, Q., and Liu, F. **Fingerprint segmentation via convolutional neural networks**. In *Chinese Conference on Biometric Recognition* (2017), Springer, pp. 324–333.

**[190]** Tang, Y., Gao, F., Feng, J., and Liu, Y. **Fingernet : An unified deep network for fingerprint minutiae extraction**. *arXiv preprint arXiv :1709.02228* (2017).

**[191]** Peralta, D., Triguero, I., García, S., Saeys, Y., Benitez, J. M., and Herrera, F. **On the use of convolutional neural networks for robust classification of multiple fingerprint captures**. *International Journal of Intelligent Systems 33*, 1 (2018), 213–230.

**[192]** Pandya, B., Cosma, G., Alani, A. A., Taherkhani, A., Bharadi, V., and McGinnity, T. **Fingerprint classification using a deep convolutional neural network**. In *2018 4th International Conference on Information Management (ICIM)* (2018), IEEE, pp. 86–91.

**[193]** Bharadi, V. A., Pandya, B., and Nemade, B. **Multimodal biometric recognition using iris & fingerprint : By texture feature extraction using hybrid wavelets**. In *Confluence The Next Generation Information Technology Summit (Confluence), 2014 5th International Conference-* (2014), IEEE, pp. 697–702.

**[194]** Yang, J., Zhang, B., and Shi, Y. **Scattering removal for finger-vein image restoration**. *Sensors 12*, 3 (2012), 3627–3640.

**[195]** Kim, H.-G., Lee, E. J., Yoon, G.-J., Yang, S.-D., Lee, E. C., and Yoon, S. M. **Illumination normalization for sift based finger vein authentication**. In *International Symposium on Visual Computing* (2012), Springer, pp. 21–30.

**[196]** Gupta, P., and Gupta, P. **An accurate finger vein based verification system**. *Digital Signal Processing 38* (2015), 43–52.

**[197]** Yang, J., and Shi, Y. **Finger-vein ROI localization and vein ridge enhancement**. *Pattern Recognition Letters 33*, 12 (2012), 1569–1579.

**[198]** Yang, J., and Yang, J. **Multi-channel gabor filter design for finger-vein image enhancement**. In *Image and Graphics, 2009. ICIG'09. Fifth International Conference on* (2009), IEEE, pp. 87–91.

**[199]** Yang, J., and Shi, Y. **Finger-vein network enhancement and segmentation**. *Pattern Analysis and Applications 17*, 4 (2014), 783–797.

**[200]** Sun, J., Jia, J., Tang, C.-K., and Shum, H.-Y. **Poisson matting**. In *ACM Transactions on Graphics (ToG)* (2004), vol. 23, ACM, pp. 315–321.

**[201]** Lee, E. C., and Park, K. R. **Image restoration of skin scattering and optical blurring for finger vein recognition**. *Optics and Lasers in Engineering 49*, 7 (2011), 816–828.

**[202]** Pi, W., Shin, J., and Park, D. **An effective quality improvement approach for low quality finger vein image**. In *Electronics and Information Engineering (ICEIE), 2010 International Conference On* (2010), vol. 1, IEEE, pp. V1–424.

**[203]** Yang, J., and Wang, J. **Finger-vein image restoration considering skin layer structure**. In *Hand-Based Biometrics (ICHB), 2011 International Conference on* (2011), IEEE, pp. 1–5.

**[204]** Xie, S. J., Lu, Y., Yoon, S., Yang, J., and Park, D. S. **Intensity variation normalization for finger vein recognition using guided filter based singe scale retinex**. *Sensors 15*, 7 (2015), 17089–17105.

**[205]** Wang, C., Peng, M., Xu, L., and Chen, T. **A single scale retinex based method for palm vein extraction**. In *Information Technology, Networking, Electronic and Automation Control Conference, IEEE* (2016), IEEE, pp. 75–78.

**[206]** He, K., Sun, J., and Tang, X. **Guided image filtering**. *IEEE transactions on pattern analysis & machine intelligence*, 6 (2013), 1397–1409.

**[207]** Fu, H., Xu, Y., Lin, S., Wong, D. W. K., and Liu, J. **Deepvessel : Retinal vessel segmentation via deep learning and conditional random field**. In *International Conference on Medical Image Computing and Computer-Assisted Intervention* (2016), Springer, pp. 132–139.

**[208]** Lafferty, J., McCallum, A., and Pereira, F. C. **Conditional random fields : Probabilistic models for segmenting and labeling sequence data**.

**[209]** Shaheed, K., Liu, H., Yang, G., Qureshi, I., Gou, J., and Yin, Y. **A systematic review of finger vein recognition techniques**. *Information 9*, 9 (2018), 213.

**[210]** Lemaitre, C., Perdoch, M., Rahmoune, A., Matas, J., and Miteran, J. **Detection and matching of curvilinear structures**. *Pattern recognition 44*, 7 (2011), 1514–1527.

**[211]** Bekkers, E., Duits, R., Berendschot, T., and ter Haar Romeny, B. **A multi-orientation analysis approach to retinal vessel tracking**. *Journal of Mathematical Imaging and Vision 49*, 3 (2014), 583–610.

**[212]** Tao, Y., and Gao, Q. **Vessel junction detection from retinal images**. In *Proceedings of the 16th International Conference on Vision Interface, Halifax, Canada* (2003), Citeseer, pp. 388–394.

**[213]** Rosdi, B. A., Shing, C. W., and Suandi, S. A. **Finger vein recognition using local line binary pattern**. *Sensors 11*, 12 (2011), 11357–11371.

**[214]** Lu, Y., Yoon, S., Xie, S. J., Yang, J., Wang, Z., and Park, D. S. **Finger vein recognition using generalized local line binary pattern.** *KSII Transactions on Internet & Information Systems 8*, 5 (2014).

**[215]** Liu, H., Song, L., Yang, G., Yang, L., and Yin, Y. **Customized local line binary pattern method for finger vein recognition**. In *Chinese Conference on Biometric Recognition* (2017), Springer, pp. 314–323.

**[216]** Trigueros, D. S., Meng, L., and Hartnett, M. **Face recognition : From traditional to deep learning methods**. *arXiv preprint arXiv :1811.00116* (2018).

**[217]** **Collection of facial images : Faces94**. Tech. rep. https://cswww.essex.ac.uk/mv/allfaces/faces94.html.

**[218]** Bonazza, P., Miteran, J., GINHAC, D., and DUBOIS, J. **Optimisation conjointe de la taille de stockage et des performances de modèles de classification pour l'authentification de visages**. In *Gretsi 2017* (2017).

**[219]** Oquab, M., Bottou, L., Laptev, I., and Sivic, J. **Learning and transferring mid-level image representations using convolutional neural networks**. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (2014), pp. 1717–1724.

**[220]** Iandola, F. N., Han, S., Moskewicz, M. W., Ashraf, K., Dally, W. J., and Keutzer, K. **Squeezenet : Alexnet-level accuracy with 50x fewer parameters and < 0.5 mb model size**.

**[221]** Zhang, X., Zhou, X., Lin, M., and Sun, J. **Shufflenet : An extremely efficient convolutional neural network for mobile devices**. *CoRR abs/1707.01083* (2017).

**[222]** Wang, R. J., Li, X., Ao, S., and Ling, C. X. **Pelee : A real-time object detection system on mobile devices**. *arXiv preprint arXiv :1804.06882* (2018).

**[223]** Freeman, I., Roese-Koerner, L., and Kummert, A. **Effnet : An efficient structure for convolutional neural networks**. *arXiv preprint arXiv :1801.06434* (2018).

**[224]** Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. **Imagenet : A large-scale hierarchical image database**. In *Computer Vision and Pattern Recognition, CVPR* (2009), pp. 248–255.

**[225]** Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., and Adam, H. **Mobilenets : Efficient convolutional neural networks for mobile vision applications**. *arXiv preprint arXiv :1704.04861* (2017).

**[226]** Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., and Chen, L.-C. **MobileNetV2 : Inverted Residuals and Linear Bottlenecks - Mobile Networks for Classification, Detection and Segmentation**.

**[227]** **Mobilenet v1 pre-trained models**. https://github.com/tensorflow/models/blob/master/research/slim/nets/mobilenet_v1.md.

**[228]** **Mobilenet v2 pre-trained models**. https://github.com/tensorflow/models/tree/master/research/slim/nets/mobilenet.

**[229]** Simonyan, K., and Zisserman, A. **Very deep convolutional networks for large-scale image recognition**. *arXiv preprint arXiv :1409.1556* (2014).

**[230]** He, K., Zhang, X., Ren, S., and Sun, J. **Deep residual learning for image recognition**. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (2016), pp. 770–778.

**[231]** Szegedy, C., Ioffe, S., Vanhoucke, V., and Alemi, A. A. **Inception-v4, inception-resnet and the impact of residual connections on learning.** In *AAAI* (2017), vol. 4, p. 12.

**[232]** Kaur, G., and Fuad, M. M. **An evaluation of protocol buffer**. In *IEEE SoutheastCon (SoutheastCon), Proceedings of the* (2010), IEEE, pp. 459–462.

**[233]** Sun, Y., Liang, D., Wang, X., and Tang, X. **Deepid3 : Face recognition with very deep neural networks**. *arXiv preprint arXiv :1502.00873* (2015).

**[234]** Parkhi, O. M., Vedaldi, A., Zisserman, A., and others. **Deep face recognition.** In *BMVC* (2015), vol. 1, British Machine Vision Association, p. 6.

**[235]** Liu, J., Deng, Y., Bai, T., Wei, Z., and Huang, C. **Targeting ultimate accuracy : Face recognition via deep embedding**. *arXiv preprint arXiv :1506.07310* (2015).

**[236]** Hambalik, P. M. **Fingerprint recognition system using artificial neural network as feature extractor : design and performance evaluation**. *Tatra Mt. Math. Publ 67* (2016), 117–134.

**[237]** Stéphan, J.-M. **Laser de faible puissance et utilisation en acupuncture : principes physiques et mécanismes d'action**. *Acupuncture et Moxibustion* (2014).

**[238]** Barolet, D. **Light-emitting diodes (leds) in dermatology**. In *Seminars in cutaneous medicine and surgery* (2008), vol. 27, No longer published by Elsevier, pp. 227–238.

**[239]** Kauba, C., Prommegger, B., and Uhl, A. **The two sides of the finger-an evaluation on the recognition performance of dorsal vs. palmar finger-veins**. In *2018 International Conference of the Biometrics Special Interest Group (BIOSIG)* (2018), IEEE, pp. 1–5.

**[240]** Zharov, V. P., Ferguson, S., Eidt, J. F., Howard, P. C., Fink, L. M., and Waner, M. **Infrared imaging of subcutaneous veins**. *Lasers in Surgery and Medicine : The Official Journal of the American Society for Laser Medicine and Surgery 34*, 1 (2004), 56–61.

**[241]** Waluś, M., Bernacki, K., and Konopacki, J. **Impact of nir wavelength lighting in image acquisition on finger vein biometric system effectiveness**. *Opto-Electronics Review 25*, 4 (2017), 263–268.

**[242]** Yin, Y., Liu, L., and Sun, X. **Sdumla-hmt : a multimodal biometric database**. In *Chinese Conference on Biometric Recognition* (2011), Springer, pp. 260–268.

**[243]** Lin, C., and Kumar, A. **Matching Contactless and Contact-Based Conventional Fingerprint Images for Biometrics Identification**. *IEEE Transactions on Image Processing 27*, 4 (2018), 2008–2021.

**[244]** Olsen, M. A., Hartung, D., Busch, C., and Larsen, R. **Contrast enhancement and metrics for biometric vein pattern recognition**. In *International Conference on Intelligent Computing* (2010), Springer, pp. 425–434.

**[245]** Buades, A., Coll, B., and Morel, J.-M. **Non-local means denoising**. *Image Processing On Line 1* (2011), 208–212.

**[246]** Buades, A., Coll, B., and Morel, J.-M. **A review of image denoising algorithms, with a new one**. *Multiscale Modeling & Simulation 4*, 2 (2005), 490–530.

**[247]** Raymond, T. **Fingerprint image enhancement and minutiae extraction**. *Report in the School of Computer Science and Software Engineering* (2003).

**[248]** Lu, Y., Wu, S., Fang, Z., Xiong, N., Yoon, S., and Park, D. S. **Exploring finger vein based personal authentication for secure iot**. *Future Generation Computer Systems 77* (2017), 149–160.

**[249]** Otsu, N. **A threshold selection method from gray-level histograms**. *IEEE transactions on systems, man, and cybernetics 9*, 1 (1979), 62–66.

**[250]** Sezgin, M., and Sankur, B. **Survey over image thresholding techniques and quantitative performance evaluation**. *Journal of Electronic imaging 13*, 1 (2004), 146–166.

**[251]** Niblack, W. **An introduction to digital image processing**, vol. 34. Prentice-Hall Englewood Cliffs, 1986.

**[252]** Van der Walt, S., Schönberger, J. L., Nunez-Iglesias, J., Boulogne, F., Warner, J. D., Yager, N., Gouillart, E., and Yu, T. **scikit-image : image processing in python**. *PeerJ 2* (2014), e453.

**[253]** Zhang, T., and Suen, C. Y. **A fast parallel algorithm for thinning digital patterns**. *Communications of the ACM 27*, 3 (1984), 236–239.

**[254]** Kawagoe, M., and Tojo, A. **Fingerprint pattern classification**. *Pattern recognition 17*, 3 (1984), 295–303.

**[255]** Calonder, M., Lepetit, V., Strecha, C., and Fua, P. **Brief : Binary robust independent elementary features**. In *European conference on computer vision* (2010), Springer, pp. 778–792.

**[256]** Pudil, P., Novovičová, J., and Kittler, J. **Floating search methods in feature selection**. *Pattern recognition letters 15*, 11 (1994), 1119–1125.

**[257]** Kuncheva, L. I. **Combining pattern classifiers : methods and algorithms**. John Wiley & Sons, 2004.

**[258]** Elhamzi, W., Dubois, J., Miteran, J., and Atri, M. **An efficient low-cost fpga implementation of a configurable motion estimation for h. 264 video coding**. *Journal of real-time image processing 9*, 1 (2014), 19–30.

**[259]** Khattab, K., Dubois, J., and Miteran, J. **Cascade boosting-based object detection from high-level description to hardware implementation**. *EURASIP Journal on Embedded Systems 2009*, 1 (2009), 235032.

**[260]** Mosqueron, R., Dubois, J., Mattavelli, M., and Mauvilet, D. **Smart camera based on embedded hw/sw coprocessor**. *EURASIP Journal on Embedded Systems 2008* (2008), 3.

**[261]** Mitéran, J., Matas, J., Bourennane, E., Paindavoine, M., and Dubois, J. **Automatic hardware implementation tool for a discrete adaboost-based decision algorithm**. *EURASIP Journal on Applied Signal Processing 2005* (2005), 1035–1046.

**[262]** Thavot, R., Mosqueron, R., Alisafaee, M., Lucarz, C., Mattavelli, M., Dubois, J., and Noel, V. **Dataflow design of a co-processor architecture for image processing**. In *Proceedings of the 2008 Conference on Design and Architectures for Signal and Image Processing* (2008), no. CONF.

**[263]** Senouci, B., Charfi, I., Heyrman, B., Dubois, J., and Miteran, J. **Fast prototyping of a soc-based smart-camera : a real-time fall detection case study**. *Journal of Real-Time Image Processing 12*, 4 (2016), 649–662.

**[264]** Marroquin, R., Dubois, J., and Nicolle, C. **Ontology for a panoptes building : Exploiting contextual information and a smart camera network**. *Semantic Web*, Preprint (2018), 1–26.

**[265]** **Intel movidius neural compute stick**. Tech. rep. https://software.intel.com/en-us/neural-compute-stick.

**[266]** Barry, B., Brick, C., Connor, F., Donohoe, D., Moloney, D., Richmond, R., O'Riordan, M., and Toma, V. **Always-on vision processing unit for mobile applications**. *IEEE Micro 35*, 2 (2015), 56–66.

**[267]** **Vpu - a dedicated vision processing platform**. Tech. rep. https://www.movidius.com/solutions/vision-processing-unit.

**[268]** **Intel movidius myriad vpu 2 : A class-defining processor**. Tech. rep. https://www.movidius.com/myriad2.

**[269]** Pena, D., Forembski, A., Xu, X., and Moloney, D. **Benchmarking of cnns for low-cost, low-power robotics applications**. In *RSS 2017 Workshop : New Frontier for Deep Learning in Robotics* (2017).

**[270]** **The gflops/w of the various machines**. Tech. rep. http://web.eece.maine.edu/~vweaver/group/green_machines.html.

**[271]** Rivas-Gomez, S., Pena, A. J., Moloney, D., Laure, E., and Markidis, S. **Exploring the vision processing unit as co-processor for inference**. In *2018 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW)* (2018), IEEE, pp. 589–598.

**[272]** Pulli, K., Baksheev, A., Kornyakov, K., and Eruhimov, V. **Real-time computer vision with opencv**. *Communications of the ACM 55*, 6 (2012), 61–69.

**[273]** Culjak, I., Abram, D., Pribanic, T., Dzapo, H., and Cifrek, M. **A brief introduction to opencv**. In *MIPRO, 2012 proceedings of the 35th international convention* (2012), IEEE, pp. 1725–1730.

**[274]** Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S., and Darrell, T. **Caffe : Convolutional architecture for fast feature embedding**. In *Proceedings of the 22nd ACM international conference on Multimedia* (2014), ACM, pp. 675–678.

**[275]** Tokui, S., Oono, K., Hido, S., and Clayton, J. **Chainer : a next-generation open source framework for deep learning**. In *Proceedings of workshop on machine learning systems (LearningSys) in the twenty-ninth annual conference on neural information processing systems (NIPS)* (2015), vol. 5, pp. 1–6.

**[276]** Seide, F., and Agarwal, A. **Cntk : Microsoft's open-source deep-learning toolkit**. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (2016), ACM, pp. 2135–2135.

**[277]** Al-Rfou, R., Alain, G., Almahairi, A., Angermueller, C., Bahdanau, D., Ballas, N., Bastien, F., Bayer, J., Belikov, A., Belopolsky, A., and others. **Theano : A python framework for fast computation of mathematical expressions**. *arXiv preprint* (2016).

**[278]** Liu, J., Dutta, J., Li, N., Kurup, U., and Shah, M. **Usability study of distributed deep learning frameworks for convolutional neural networks**.

**[279]** Shi, S., Wang, Q., Xu, P., and Chu, X. **Benchmarking state-of-the-art deep learning software tools**. In *Cloud Computing and Big Data (CCBD), 2016 7th International Conference on* (2016), IEEE, pp. 99–104.

**[280]** Bahrampour, S., Ramakrishnan, N., Schott, L., and Shah, M. **Comparative study of deep learning software frameworks**. *arXiv preprint arXiv :1511.06435* (2015).

**[281]** Nakaya, H., Komiyama, Y., Kashikawa, N., Uchida, T., Nagayama, T., and Yoshida, M. **Low temperature performance of a commercially available ingaas image sensor**. In *High Energy, Optical, and Infrared Detectors for Astronomy VII* (2016), vol. 9915, International Society for Optics and Photonics, p. 99151O.

**[282]** Houver, S., Cavalié, P., St-Jean, M. R., Amanti, M., Sirtori, C., Li, L., Davies, A., Linfield, E., Pereira, T., Lebreton, A., and others. **Optical sideband generation up to room temperature with mid-infrared quantum cascade lasers**. *Optics express 23*, 4 (2015), 4012–4020.

**[283]** Houver, S., Lebreton, A., Mottaghizadeh, A., Amanti, M. I., Sirtori, C., Beaudoin, G., Sagnes, I., Parillaud, O., Colombelli, R., Mangeney, J., and others. **Multi-terahertz sideband generation on an optical telecom carrier with a quantum cascade laser**. *ACS Photonics 5*, 3 (2018), 890–896.

**[284]** Prados, E., and Faugeras, O. **Shape from shading**. In *Handbook of mathematical models in computer vision*. Springer, 2006, pp. 375–388.

**[285]** Durou, J.-D., Falcone, M., and Sagona, M. **A survey of numerical methods for shape from shading**. *Rapport de recherche 2* (2004).

**[286]** Jiang, L., Zhang, J., Deng, B., Li, H., and Liu, L. **3d face reconstruction with geometry details from a single image**. *IEEE Transactions on Image Processing 27*, 10 (2018), 4756–4770.

**[287]** Bobbia, S., Benezeth, Y., and Dubois, J. **Remote photoplethysmography based on implicit living skin tissue segmentation**. In *2016 23rd International Conference on Pattern Recognition (ICPR)* (2016), IEEE, pp. 361–365.

**[288]** Bobbia, S., Luguern, D., Benezeth, Y., Nakamura, K., Gomez, R., and Dubois, J. **Real-time temporal superpixels for unsupervised remote photoplethysmography**. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops* (2018), pp. 1341–1348.

**[289]** Bonazza, P., Miteran, J., Ginhac, D., and Dubois, J. **Comparative study of deep learning and classical methods : smart camera implementation for face authentication**. In *Counterterrorism, Crime Fighting, Forensics, and Surveillance Technologies II* (2018), vol. 10802, International Society for Optics and Photonics, p. 108020I.

**[290]** Bonazza, P., Mitéran, J., Ginhac, D., and Dubois, J. **Machine learning vs transfer learning smart camera implementation for face authentication**. In *Proceedings of the 12th International Conference on Distributed Smart Cameras* (2018), ACM, p. 21.

**[291]** Bonazza, P., Mitéran, J., Heyrman, B., Ginhac, D., Thivent, V., and Dubois, J. **An affordable contactless security system access for restricted area**. In *5th Workshop on Architecture of Smart Cameras (WASC)* (2016).

**[292]** Dubois, J., Moinet, A., Bobbia, S., Marroquin, R., Heyrman, B., Bonazza, P., Darties, B., Nicolle, C., Benezeth, Y., Mitéran, J., and others. **Wiseeye : A platform to manage and experiment on smart camera networks**. In *5th Workshop on Architecture of Smart Cameras (WASC)* (2016).

**[293]** Kataoka, H., Iwata, K., and Satoh, Y. **Feature evaluation of deep convolutional neural networks for object recognition and detection**. *arXiv preprint arXiv :1509.07627* (2015).

**[294]** Jeong, C. B., Kim, K. G., Kim, T. S., and Kim, S. K. **Comparison of image enhancement methods for the effective diagnosis in successive whole-body bone scans**. *Journal of digital imaging 24*, 3 (2011), 424–436.

**[295]** Li, H., Xu, Z., Taylor, G., Studer, C., and Goldstein, T. **Visualizing the loss landscape of neural nets**. In *Advances in Neural Information Processing Systems* (2018), pp. 6389–6399.

**[296]** Pizer, S. M., Amburn, E. P., Austin, J. D., Cromartie, R., Geselowitz, A., Greer, T., ter Haar Romeny, B., Zimmerman, J. B., and Zuiderveld, K. **Adaptive histogram equalization and its variations**. *Computer vision, graphics, and image processing 39*, 3 (1987), 355–368.

**[297]** Zuiderveld, K. **Contrast limited adaptive histogram equalization**. In *Graphics gems IV* (1994), Academic Press Professional, Inc., pp. 474–485.

**[298]** Zimmerman, J. B., Cousins, S. B., Hartzell, K. M., Frisse, M. E., and Kahn, M. G. **A psychophysical comparison of two methods for adaptive histogram equalization**. *Journal of Digital Imaging 2*, 2 (1989), 82.

**[299]** Awad, A. I., and Baba, K. **Singular point detection for efficient fingerprint classification**. *International Journal of New Computer Architectures and Their Applications 2*, 1 (2012), 1–8.

**[300]** Bernsen, J. **Dynamic thresholding of gray-level images**. In *Proc. Eighth Int'l conf. Pattern Recognition, Paris, 1986* (1986).

**[301]** Sauvola, J., and Pietikäinen, M. **Adaptive document image binarization**. *Pattern recognition 33*, 2 (2000), 225–236.

**[302]** Phansalkar, N., More, S., Sabale, A., and Joshi, M. **Adaptive local thresholding for detection of nuclei in diversity stained cytology images**. In *Communications and Signal Processing (ICCSP), 2011 International Conference on* (2011), IEEE, pp. 218–220.

**[303]** Zou, K. H., Warfield, S. K., Bharatha, A., Tempany, C. M., Kaus, M. R., Haker, S. J., Wells III, W. M., Jolesz, F. A., and Kikinis, R. **Statistical validation of image segmentation quality based on a spatial overlap index1 : scientific reports**. *Academic radiology 11*, 2 (2004), 178–189.

# Table des figures

# Liste des tables

# Liste des acronymes

**ACI *(ICA)* :** **Analyse en Composantes Indépendantes - *Independant Component Analysis***

**ACP *(PCA)* :** **Analyse en Composantes Principales - *Principal Component Analysis***

**ADL *(LDA)* :** **Analyse Discriminante Linéaire - *Linear Discriminent Analysis***

**AIEA :** **Agence Internationale de l'Énergie Atomique**

***ANN* :** **Réseau de Neurones Artificiels - *Artificial Neural Network***

***ARM* :** architecture de processeurs - ***Ashton Raggatt McDougall***

**ASN :** **Agence de Sureté Nucléaire**

***BRIEF* :** ***Binary Robust Independent Elementary Features***

***CCD* :** ***Charge Coupled Device***

***CLAHE* :** ***Contrast-Limited Adaptive Histogram Equalization***

***CMOS* :** ***Complementary Metal Oxide Semiconductor***

***sCMOS* :** ***scientific Complementary Metal Oxide Semiconductor***

**CNIL :** **Commission Nationale de l'Informatique et des Libertés**

***CNN* :** ***Convolutional Neural Network***

***CPU* :** ***Central Processing Unit***

***DCNN* :** ***Deep Convolutional Neural Network***

***DL* :** ***Deep Learning***

***FN / FP* :** **Faux Négatifs / Faux Positifs - *False Negative / False Positive***

**FUI :** **Fonds Unique Interministériel**

***GPU* :** ***Graphics Processing Unit***

**IHM :** **Interface Homme-Machine**

**ImViA :** **laboratoire Imagerie et Vision Artificielle**

***InGaAs*** **:** ***Indium Gallium Arsenide***

**ips** ***(fps)*** **:** **images par secondes -** ***frame per second***

**IRSN :** **Institut de Radioprotection et de Sûreté Nucléaire**

**Ju** ***(Ac)*** **:** **Justesse de classification -** ***Accuracy***

***KNN*** **:** ***k-Nearest Neighbors***

**Ko/Mo :** **Kilooctets / Mégaoctets**

**Le2i :** **Laboratoire Électronique Informatique et Image**

***LFW*** **:** ***Labeled Faces in the Wild database***

**Lp / L1 / L2 :** **Distance de Minkowski / Distance de Mannhatan / Distance Euclidienne**

***MACs*** **:** ***Multiply-ACcumulate operations***

***ML*** **:** ***Machine Learning***

**ONU :** **Organisation des Nations Unies**

***PC*** **:** ***Personal Computer***

**Pr :** **Précision de classification**

**RB :** **Réseau Bayesien**

***RBF*** **:** ***Radial Basis Function***

***RF*** **:** ***Random Forest***

***RFID*** **:** ***Radio Frequency Identification***

***RNN*** **:** ***Recurrent Neural Networks***

**RPP :** **Réduction de Précision de Paramètres**

**s / ms / µs :** **seconde, milliseconde, microseconde**

**Se :** **Sensibilité (ou rappel) de classification**

**Sp :** **Spécificité de classification**

***SBFS*** **:** ***Sequential Backward Floating Selection***

***SIFT*** **:** ***Scale-Invariant Feature Transform***

***SURF*** **:** ***Speeded-Up Robust Features***

**SVM :** **Séparateurs à Vaste Marge -** ***Support Vector Machine***

**VN / VP** ***(TN / TP)*** **:** **Vrais Négatifs / Vrais Positifs -** ***True Negative / True Positive***

***VPU*** **:** ***Vision Processing Unit***

# Annexes

# A

# Précisions sur les algorithmes

## Sommaire

## A.1/ Machine Learning Classique

### A.1.1/ Analyse en Composantes Principales (ACP)

Soit un ensemble de $M$ images de dimension [$m$ x $n$], nommées $I_{i_{(m,n)}}$. Celles-ci sont alors traitées comme un vecteur $\Gamma_{i_{(N,1)}}$ dans un espace vectoriel de grande dimension [$N = m$ x $n$], par concaténation des colonnes de pixels de l'image, selon l'équation A.1. Les éléments de la matrice $I_i$ et du vecteur $\Gamma_i$ représentent les pixels de l'image $I_i$ en niveau de gris, codés de 0 à 255.

$$I_i = \begin{bmatrix} a_{1,1} & \cdots & a_{1,m} \\ \vdots & \ddots & \vdots \\ a_{n,1} & \cdots & a_{n,m} \end{bmatrix} \quad \Rightarrow \quad \Gamma_i = \begin{bmatrix} a_{1,1} \\ \vdots \\ a_{n,1} \\ \vdots \\ a_{1,m} \\ \vdots \\ a_{n,m} \end{bmatrix} \tag{A.1}$$

Les image $I_i$ vectorisées en un vecteur $\Gamma_i$ sont alors regroupées dans une unique matrice $\Gamma$, où chaque colonne représente un vecteur $\Gamma_i$, tel que le montre l'égalité suivante :

$$\Gamma = \begin{bmatrix} \Gamma_1 & \cdots & \Gamma_j & \cdots \Gamma_M \end{bmatrix} = \begin{bmatrix} a_{1_{1,1}} & \cdots & a_{j_{1,1}} & \cdots & a_{M_{1,1}} \\ \vdots & & \vdots & & \vdots \\ a_{1_{n,1}} & \cdots & a_{j_{n,1}} & \cdots & a_{M_{n,1}} \\ \vdots & & \vdots & & \vdots \\ a_{1_{1,m}} & \cdots & a_{j_{1,m}} & \cdots & a_{M_{1,m}} \\ \vdots & & \vdots & & \vdots \\ a_{1_{n,m}} & \cdots & a_{j_{n,m}} & \cdots & a_{M_{n,m}} \end{bmatrix}$$

Puis un nouveau vecteur, $\Psi$, représentant la moyenne des images vectorisée, est alors calculé selon l'équation A.2. Ce calcul permet ensuite, selon l'équation A.3 de normaliser chaque vecteur $\Gamma_i$ en leur soustrayant $\Psi$ pour alors obtenir un vecteur normalisé $\Phi_i$.

$$\Psi = \frac{1}{M} \sum_{i=1}^{M} \Gamma_i \tag{A.2}$$

$$\Phi_i = \Gamma_i - \Psi, \qquad \forall i \in [1; M] \tag{A.3}$$

La matrice de covariance $C$ de cet ensemble de nouvelles données $\Phi$ est alors calculée, selon l'équation A.4, où $\Phi_i^T$ correspond au transposé du vecteur $\Phi_i$. C'est à partir de cette matrice, de dimension ($N$ x $N$), soit ($m.n$ x $m.n$), que sont ensuite calculés les vecteurs propres et les valeurs propres, représentant les caractéristiques de l'ensemble des $M$ images d'origine.

$$\begin{cases} C = \sum_{i=1}^{M} \Phi_i \Phi_i^T \\ C = AA^T, \quad \text{où } A = [\Phi_1 \Phi_2 \cdots \Phi_M] \end{cases} \tag{A.4}$$

Considérons les vecteurs propres $e_i$ de $C = AA^T$, associés aux valeurs propres $\lambda_i$. L'équation A.5 établi la relation entre ces différents éléments.

$$\begin{cases} Ce_i &= \lambda_i e_i \\ AA^T e_i &= \lambda_i e_i \end{cases} \tag{A.5}$$

En l'état, ce calcul représente une très grande complexité, de l'ordre du nombre de pixels dans l'image (pour une image de dimension 100x100, la matrice de covariance est de dimension 10000x10000). Or, dans le cas où la définition $N$ (nombre de pixels) est supérieure au nombre d'images $M$, seuls $M - 1$ vecteurs propres contiendront de l'information, les autres auront des valeurs propres associées nulles. Il s'agit alors de transformer le problème actuel pour atteindre une complexité de l'ordre du nombre d'images $M$. Pour ce faire, une nouvelle matrice $L = A^T A$, de dimension MxM, est introduite. Considérons ses vecteurs propres $v_i$ et les valeurs propres $\mu_i$ associées, l'équation A.6 établi la relation entre ces différents éléments.

$$\begin{cases} Lv_i &= \mu_i v_i \\ A^T A v_i &= \mu_i v_i \end{cases} \tag{A.6}$$

D'après l'équation A.6, en multipliant par $A$ chaque côté de l'égalité, on obtient :

$$AA^T A v_i = A\mu_i v_i \tag{A.7}$$

Cependant, comme $C = AA^T$, on peut simplifier l'égalité A.7 et obtenir l'équation A.8 :

$$C(Av_i) = \mu_i(Av_i) \tag{A.8}$$

Ainsi, d'après les équations A.5 et A.8, on observe que $Av_i$ et $\mu_i$ sont respectivement les valeurs propres de C, on obtient alors :

$$\begin{cases} e_i &= Av_i \\ \lambda_i &= \mu_i \end{cases} \tag{A.9}$$

L'équation A.9 montre que les valeurs propres de la matrice $C$ peuvent être calculées en trouvant les valeurs propres de la matrice $L$. Pour trouver les vecteurs propres de $C$, il suffit de multiplier les vecteurs propres de $L$ par la matrice $A$. Par la suite, les vecteurs propres trouvés sont ordonnés suivant l'ordre décroissant de leur valeurs propres associées, permettant alors d'obtenir des informations sur ces couples vecteurs propres. En effet, plus la valeur propre est grande, plus variance capturée par le vecteur propre est importante, traduisant alors une relation entre la quantité d'informations contenue dans ce vecteur et sa valeur propre. L'espace vectoriel engendré par l'ensemble de ces vecteurs propres est alors noté $E_v$ et permet, en y projetant une nouvelle image, de la décrire avec les informations (caractéristiques) de cet espace. La moyenne $\Psi$ est soustraite à cette nouvelle image, que l'on vectorise en $\Gamma$, et les poids $\omega_k$ décrivant cette nouvelle image en fonction des vecteurs propres $e_k$ de l'espace $E_v$ sont donnés par l'équation A.10. Ces poids forment alors une matrice $\Omega^T = [\omega_1, \omega_2, ..., \omega_M]$, décrivant au mieux la nouvelle image selon les caractéristiques de l'ensemble ayant créé l'espace $E_v$

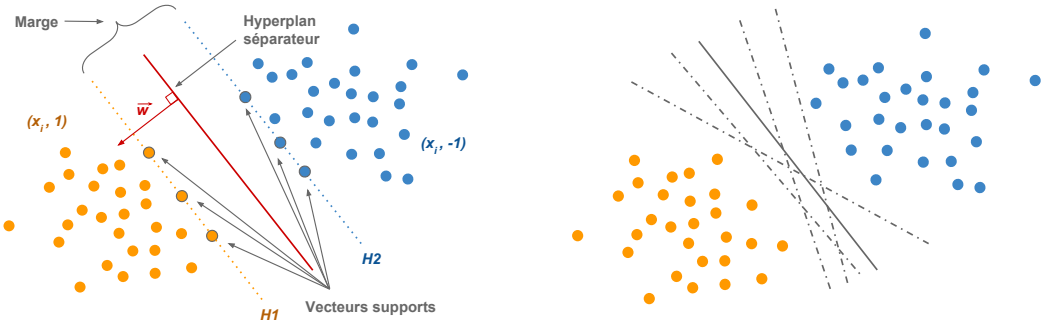
$$\omega_k = e_k^T(\Gamma - \Psi), \quad \text{où } k = 1, ..., M \tag{A.10}$$

### A.1.2/ Machines à vecteurs supports (SVM)

#### A.1.2.1/ Détermination des hyperplans des SVM

Les Séparateurs à Vastes Marges, encore appelés Machines à Vecteurs Supports (SVM), cherchent à séparer au mieux des ensembles de données appartenant à des classes. Cette séparation, comme illustrée par la figure A.1a, est effectuée par un hyperplan (ou une droite dans un plan en deux dimensions). Celui-ci est généré, ou supporté, par certains vecteurs appartenant aux classes à séparer, que l'on appelle alors vecteurs supports. Pour chaque classe, un hyperplan "support" passe par ces vecteurs supports et l'hyperplan séparateur est situé à égale distance de chacun d'eux. Cette distance est appelée "la marge". Il existe une infinité de ligne ou d'hyperplan séparant des données linéairement séparables, hors le but est d'effectuer une classification et donc d'être capable de prédire, avec une erreur minimale, l'appartenance d'une nouvelle donnée à telle ou telle classe. Ainsi, l'objectif des SVM est de trouver l'hyperplan maximisant la marge et donc séparant au mieux les classes et minimisant l'erreur. Comme le montre la figure A.1b, plusieurs hyperplans sont générés à partir de divers vecteurs supports, et l'hyperplan fournissant la marge maximale est alors retenu.

**Séparateurs linéaires - Cas linéairement séparable :**




(a) Principe de la marge des SVM (b) Recherche d'un hyperplan maximisant la marge

Figure A.1 – Séparations de classes par les SVM dans le cas de données linéairement séparables

Considérons un ensemble de $N$ données de dimensions $n$ (i.e N vecteurs chacun de $n$ éléments) nommées $\vec{x_i}$, où $i \in [1; N]$. Ces données sont réparties selon deux classes nommées $y_i$ telles que $y_i \in \{+1, -1\}$ et sont représentés par le couple $(x_i, yi)$. Soient le vecteur $\vec{w}$ représentant la normale de l'hyperplan séparateur, tel que $\vec{w} = (w_1, ..., w_n)$, et $b$ la constante de biais. L'hyperplan $H$ séparant ces données, et les hyperplans supports $H_1$ et $H_2$ vérifient l'équation suivante :

$$f(\vec{x_i}) = y_i(\vec{w}.\vec{x_i} + b) \geq 1 \tag{A.11}$$

Cette équation signifie que tous les points d'une même classe sont du même côté de l'hyperplan, le signe de $f(x_i)$ donnant la classe associée du vecteur $x_i$. Ces hyperplans ont alors pour équation :

$$\begin{cases} H_1 & : \vec{w}.\vec{x_i} + b = 1 \\ H & : \vec{w}.\vec{x_i} + b = 0 \\ H_2 & : \vec{w}.\vec{x_i} + b = -1 \end{cases} \tag{A.12}$$

La distance entre un vecteur $\vec{x}$ et l'hyperplan $H$, et la marge $M$, sont définis par :

$$d(\vec{x}, H) = \frac{f(\vec{x})}{\|\vec{w}\|} \qquad M = \frac{2}{\|\vec{w}\|} \tag{A.13}$$

Déterminer l'hyperplan optimal équivaut à maximiser la marge $M$ et revient donc à minimiser $\|\vec{w}\|$. C'est un problème d'optimisation quadratique, appelé formulation primale, et peut être résolu grâce à la méthode de Lagrange. Le lagrangien, $L$, est formé en pondérant la contrainte par $\vec{\alpha}$ (composé de $l$ multiplicateurs de Lagrange $\alpha_i$, pour chacun des $l$ vecteurs $x_i$ vérifiant l'équation A.11), et en la soustrayant à la fonction de coût $\Phi$ :

$$L(\vec{w}, b, \vec{\alpha}) = \frac{1}{2}\|\vec{w}\|^2 - \sum_{i=1}^{l} \alpha_i [y_i(\vec{w}.\vec{x_i} + b) - 1] \quad \begin{cases} Minimiser \;\; \Phi = \dfrac{\|\vec{w}\|^2}{2} \\ f(x_i) = y_i(\vec{w}.\vec{x_i} + b) \geq 1 \\ \alpha_i \leq 0 \quad \forall i \in [1; l] \end{cases} \tag{A.14}$$

Les solutions sont définies comme optimales suivant deux conditions optimales dépendant des dérivées partielles du Lagrangien par rapport à $\vec{w}$ et à b :

$$\begin{cases} \text{Condition 1 :} & \dfrac{\partial L(\vec{w}, b, \vec{\alpha})}{\partial \vec{w}} = 0 & \Rightarrow & \vec{w} = \displaystyle\sum_{i=1}^{l} \alpha_i y_i \vec{x_i} \\ \text{Condition 2 :} & \dfrac{\partial L(\vec{w}, b, \vec{\alpha})}{\partial b} = 0 & \Rightarrow & \displaystyle\sum_{i=1}^{l} \alpha_i y_i = 0 \end{cases} \tag{A.15}$$

En développant le Lagrangien et en se plaçant à ces conditions optimales, on obtient alors :

$$\begin{cases} L(\vec{w}, b, \vec{\alpha}) = \dfrac{1}{2}\|\vec{w}\|^2 - \displaystyle\sum_{i=1}^{l} \alpha_i y_i \vec{w}.\vec{x_i} - b \sum_{i=1}^{l} \alpha_i y_i + \sum_{i=1}^{l} \alpha_i \\ \dfrac{1}{2}\|\vec{w}\|^2 = \displaystyle\sum_{i=1}^{l}\sum_{j=1}^{l} \alpha_i \alpha_j y_i y_j \vec{x_i}.\vec{x_j} \end{cases} \Rightarrow \; L(\vec{\alpha}) = \sum_{i=1}^{l} \alpha_i + \sum_{i=1}^{l}\sum_{j=1}^{l} \alpha_i \alpha_j y_i y_j \vec{x_i}.\vec{x_j} \tag{A.16}$$

Les $\alpha_{i,0}$ sont les multiplicateurs optimaux associés aux vecteurs d'apprentissage $x_i$. Les $x_i$ associés aux $\alpha_{i,0}$ non nuls sont alors les vecteurs supports, notés $S_i$, et sont situés sur $H_1$ et $H_2$. Tous les autres vecteurs sont associés à un $\alpha_{i,o}$ nul et sont situées de chaque côté de ces hyperplans. Le vecteur de poids optimal $\vec{w_o}$ permet de définir l'équation de l'hyperplan optimal $H_{opt}$ :

$$\vec{w_o} = \sum_{i=1}^{l} \alpha_{i,o} y_i \vec{S_i} \quad \Rightarrow \quad H_{opt} : \vec{w_o}\vec{x} + b_o = \sum_{i=1}^{l} \alpha_{i,o} y_i \vec{S_i}.\vec{x} + b_o = 0 \tag{A.17}$$

De manière similaire, pour un nouvel échantillon $x^*$, la fonction de décision devient :

$$g(\vec{x^*}) = signe\left(\sum_{i=1}^{l} \alpha_{i,o} y_i \vec{S_i}.\vec{x^*} + b_o\right) \tag{A.18}$$

**Séparateurs linéaires - Cas non linéairement séparable :**
Dans des cas où, tel qu'illustré par la figure 3.4b, les données ne sont pas linéairement séparables dû à du bruit comme dans la plupart des problèmes réels, la notion de marge souple a été introduite. En effet, la définition de la marge telle qu'elle a été présentée précédemment ne peut pas s'appliquer du fait de la répartition des données (le barycentre de chaque classe est bien distinct mais la séparation n'est pas clairement définissable). Ainsi, aux formules précédentes s'ajoute un ensemble de variables d'ajustement, "$\epsilon_i$" (marge d'erreur), contrôlées par un paramètre de régularisation, "$C$", étant une constante pondérant cette erreur. La formulation de l'équation A.11 d'un hyperplan, et la fonction "$\Phi$" à minimiser pour trouver l'hyperplan optimal, au travers de la méthode de Lagrange à l'équation A.14, deviennent :

$$\begin{cases} Minimiser \;\; \Phi = \dfrac{\|\vec{w}\|^2}{2} + C \sum_i^l \epsilon_i \\ Tel\ que \;\; f(x_i) = y_i(\vec{w}.\vec{x_i} + b) \geq 1 - \epsilon_i \end{cases} \tag{A.19}$$

Le paramètre $\epsilon_i$ défini la localisation de la donnée "i" par rapport à la marge, tel que le montre la figure A.2. Une valeur à zéro établi que l'observation est du bon côté de la marge ou sur la marge, une valeur supérieure à zéro indique que l'observation est du mauvais côté et une valeur supérieure à un défini une donnée du mauvais côté de l'hyperplan. Le paramètre $C$ contrôle le compromis entre la largeur de marge et les erreurs de classification. Par conséquent, plus $C$ sera petit et moins les erreurs seront prises en compte (plus la marge sera large). Plus $C$ sera grand, plus les erreurs de classification seront considérées comme importante (plus la marge sera restreinte).

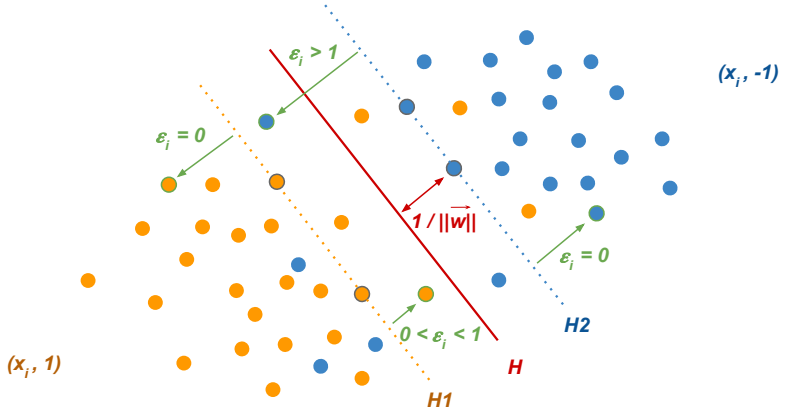



FIGURE A.2 – Séparation de classes par les SVM dans le cas de données non linéairement séparables

**Séparateurs non linéaires :**
Dans la majorité des cas en application réelle, tel qu'illustré par la figure 3.4c, le problème de répartition de données est plus complexe et la frontière ne peut être linéaire. Pour répondre à une telle problématique, une fonction noyau non linéaire, "$K(x, y)$", est introduite, permettant de représenter les données dans un nouvel espace où il est possible de les séparer par un hyperplan, tel que l'illustre la figure A.3.

En appliquant cette fonction noyau "$K$" au problème d'optimisation quadratique pour des vecteurs supports "$S_i$" formant l'hyperplan optimal, le Lagrangien devient :

$$\begin{cases} Minimiser \ \ \Phi = \dfrac{\|\vec{w}\|^2}{2} + C\sum_{i}^{l}\epsilon_i \\ L(\vec{\alpha}) = \sum_{i=1}^{l}\alpha_i + \sum_{i=1}^{l}\sum_{j=1}^{l}\alpha_i\alpha_j y_i y_j K(\vec{x_i}.\vec{x_j}) - C\sum_{i}^{l}\epsilon_i \end{cases} \tag{A.20}$$

La fonction de classification finale d'un nouvel échantillon $x^*$, avec la fonction noyau $K$, devient :

$$g(\vec{x^*}) = signe(\sum_{i=1}^{l}\alpha_{i,o}y_i K(\vec{S_i}.\vec{x^*}) + b_o) \tag{A.21}$$

La fonction noyau $K(x, y)$ peut prendre différentes formes, et le choix de cette forme est généralement effectué de manière empirique selon l'application. Chacune de ces formes introduit de nouveau paramètres, ($\gamma$, $\theta$, $q$), qui sont eux aussi à optimiser lors de l'apprentissage. Certains noyaux sont plus utilisés que d'autres dans la littérature, les trois principaux étant :

— **Fonction de base radiale gaussienne *(gaussian Radial Basis Function - RBF)* :**

$$K(\vec{x}, \vec{y}) = e^{(\gamma\|\vec{x}-\vec{y}\|^2)} \tag{A.22}$$

— **Noyau Polynomial (d'ordre "$q$") :**

$$K(\vec{x}, \vec{y}) = [\gamma(\vec{x}.\vec{y}) + \theta]^q \tag{A.23}$$

— **Noyau Sigmoïde ou tangente hyperbolique :**

$$K(\vec{x}, \vec{y}) = tanh[\gamma(\vec{x}.\vec{y}) - \theta] \tag{A.24}$$

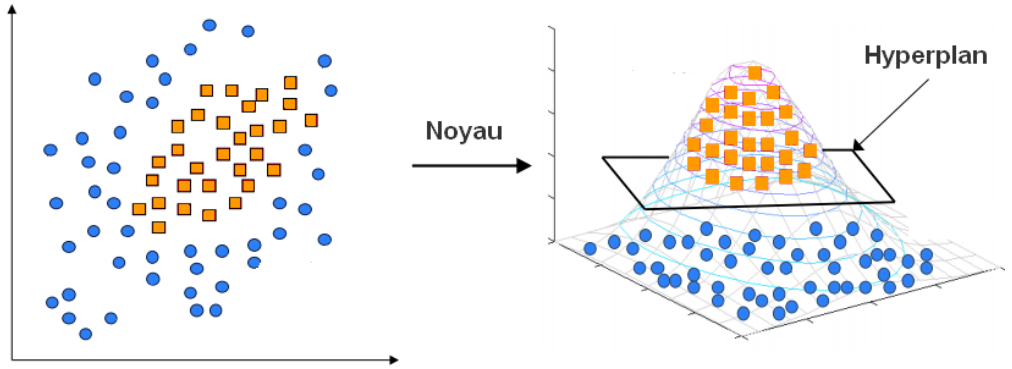



FIGURE A.3 – Noyau appliqué aux SVM dans le cas de données non linéairement séparables[1]

1. Image source : hackerearth.com - "simple tutorial svm parameter tuning"

### A.1.3/ Fonctionnement des classifieurs

#### A.1.3.1/ Exemple d'un modèle de classification Random Forest

```
<?xml version="1.0"?>
<opencv_storage>
<my_random_trees type_id="opencv-ml-random-trees">
  <nclasses>2</nclasses>
  <nsamples>80</nsamples>
  <nactive_vars>10</nactive_vars>
  <oob_error>0.</oob_error>
  <ntrees>39</ntrees>
  <is_classifier>1</is_classifier>
  <var_all>112</var_all>
  <var_count>112</var_count>
  <ord_var_count>112</ord_var_count>
  <cat_var_count>0</cat_var_count>
  <training_params>
       <use_surrogates>1</use_surrogates>
       <max_categories>10</max_categories>
       <max_depth>4</max_depth>
       <min_sample_count>2</min_sample_count>
       <cross_validation_folds>0</cross_validation_folds>
       <priors type_id="opencv-matrix">
         <rows>1</rows>
         <cols>2</cols>
         <dt>d</dt>
         <data>
              1. 1.</data></priors></training_params>
  <var_type>
       0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
       0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
       0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
       0 0 0 0 0 0 0 0 0 0</var_type>
  <cat_count type_id="opencv-matrix">
       <rows>1</rows>
       <cols>1</cols>
       <dt>i</dt>
       <data>
         2</data></cat_count>
  <cat_map type_id="opencv-matrix">
       <rows>1</rows>
       <cols>2</cols>
       <dt>i</dt>
       <data>
         0 1</data></cat_map>
  <trees>
       <_>
         <best_tree_idx>-1</best_tree_idx>
         <nodes>
              <_>
                <depth>0</depth>
                <sample_count>80</sample_count>
                <value>0.</value>
                <norm_class_idx>0</norm_class_idx>
                <Tn>0</Tn>
                <complexity>0</complexity>
                <alpha>0.</alpha>
```

```
      <node_risk>38.</node_risk>
      <tree_risk>0.</tree_risk>
      <tree_error>0.</tree_error>
      <splits>
        <_><var>20</var>
          <quality>6.4062538146972656e+01</quality>
          <le>3.7914166259765625e+02</le></_></splits></_>
    <_>
        [...]
    <_>
      <depth>3</depth>
      <sample_count>36</sample_count>
      <value>1.</value>
      <norm_class_idx>1</norm_class_idx>
      <Tn>0</Tn>
      <complexity>0</complexity>
      <alpha>0.</alpha>
      <node_risk>1.</node_risk>
      <tree_risk>0.</tree_risk>
      <tree_error>0.</tree_error>
      <splits>
        <_><var>87</var>
          <quality>36.</quality>
          <le>4.4903387451171875e+02</le></_></splits></_>
    <_>
      <depth>4</depth>
      <sample_count>1</sample_count>
      <value>0.</value>
      <norm_class_idx>0</norm_class_idx>
      <Tn>0</Tn>
      <complexity>0</complexity>
      <alpha>0.</alpha>
      <node_risk>0.</node_risk>
      <tree_risk>0.</tree_risk>
      <tree_error>0.</tree_error></_></nodes></_></trees></my_random_trees>
</opencv_storage>
```

### A.1.3.2/ Exemple d'un modèle de classification Machines à Vecteurs Supports

```
<?xml version="1.0"?>
<opencv_storage>
<my_svm type_id="opencv-ml-svm">
  <svm_type>C_SVC</svm_type>
  <kernel><type>RBF</type>
       <gamma>1.4901161193847656e-08</gamma></kernel>
  <C>5.0000000000000000e-01</C>
  <term_criteria><epsilon>1.0000000474974513e-03</epsilon>
       <iterations>2147483647</iterations></term_criteria>
  <var_all>112</var_all>
  <var_count>112</var_count>
  <class_count>2</class_count>
  <class_labels type_id="opencv-matrix">
       <rows>1</rows>
       <cols>2</cols>
       <dt>i</dt>
       <data>
         0 1</data></class_labels>
  <sv_total>52</sv_total>
  <support_vectors>
       <_>
         -4.49642700e+02 1.01862000e+03 -1.51037720e+03 -4.57064697e+02
         -9.26060364e+02 -1.35186523e+02 -5.23293762e+02 -2.68006470e+02
         -9.30458374e+02 -1.26954071e+02 -1.22082198e+00 1.30249069e+02
         3.55707428e+02 1.72554993e+02 -3.93793091e+02 3.02044468e+01
         -6.41788574e+02 -1.10044304e+02 1.26496094e+02 3.89551277e+01
         1.32348495e+02 8.65020523e+01 6.86631393e+01 -7.93158951e+01
         1.99552948e+02 2.27943069e+02 4.24283630e+02 4.74550385e+02
         9.94194717e+01 -3.16993530e+02 -5.02721901e+01 -3.45211456e+02
         -3.68552132e+01 -4.83171158e+01 4.05100586e+02 -1.73725567e+01
         -4.83093842e+02 -3.83693542e+02 1.92268295e+02 2.32956726e+02
         1.36045361e+00 2.06012787e+02 -5.75903320e+01 -3.93775391e+02
         2.53923401e+02 -1.91929016e+01 1.65237930e+02 3.40357933e+01
         -2.41368103e+02 2.55680420e+02 -1.56458984e+02 6.14017715e+01
         1.08755943e+02 2.67144043e+02 -1.64698906e+01 1.26423286e+02
         2.16332123e+02 -1.74757416e+02 -1.15382057e+02 2.15967957e+02
         9.07019196e+01 1.79278549e+02 -6.06359634e+01 4.25044281e+02
         -1.62669754e+01 -1.08339378e+02 -1.64651474e+02 -2.60441162e+02
         4.23969238e+02 1.10181526e+02 2.58136200e+02 1.05583481e+02
         5.79773102e+01 1.18944885e+02 -1.54037750e+02 -1.28837982e+02
         -1.01631287e+02 -1.62281769e+02 -1.14152824e+02 7.71475296e+01
         -1.04492500e+02 -3.18244667e+01 -5.42519836e+01 -1.14186684e+02
         -1.79596756e+02 1.17646751e+02 3.30134926e+01 1.06564247e+02
         2.07701797e+02 -2.52759399e+01 2.26194000e+02 7.81666641e+01
         -2.07448978e+01 1.13684776e+02 1.04179945e+01 -9.92391014e+00
         9.47137833e+01 -7.25973740e+01 1.31979080e+02 1.51288330e+02
         -3.46075630e+01 -8.10010147e+01 -1.31766556e+02 -1.01447632e+02
         -3.61141510e+01 1.42644104e+02 -2.01892681e+01 1.18168282e+02
         1.61334534e+02 -2.29486221e+02 -1.14929626e+02 -7.13307953e+01</_>
       <_>
            [...]
       <_>
         3.15939355e+03 3.68422144e+03 -3.68870049e+01 -9.70469543e+02
         -1.79883496e+03 7.03784851e+02 1.08469948e+02 -1.46755676e+03
         6.15075317e+02 -2.05773950e+03 -1.14626646e+01 1.23982407e+02
         6.74557922e+02 1.59165515e+03 -7.65604858e+01 -5.74217285e+02
         -7.68174820e+01 -4.48738556e+02 -2.43399567e+02 -3.21951141e+02
```

```
        1.70471008e+03 4.71126862e+02 4.64464386e+02 1.32299795e+01
        4.91791229e+02 -1.81619431e+02 3.21018646e+02 3.94631592e+02
        3.24940430e+02 -4.67548141e+01 -6.46302673e+02 5.21366211e+02
        6.22378357e+02 1.45610214e+02 7.66079895e+02 -6.19102173e+02
        -6.72862244e+02 -2.29719101e+02 -7.26887131e+01 8.88097900e+02
        5.97086975e+02 -6.09501648e+02 -2.45164413e+02 -1.43837051e+02
        3.26460934e+00 3.40058044e+02 4.45278664e+01 3.09685822e+01
        -7.21426086e+01 -2.58392609e+02 2.83273926e+02 -1.30227509e+02
        4.91925903e+02 -3.62057190e+01 -1.81259995e+01 5.23059570e+02
        -3.02842163e+02 5.16690750e+01 -7.23259521e+02 -2.59948029e+02
        1.12777739e+01 -2.73808380e+02 7.85895874e+02 1.89494492e+02
        -1.61327759e+02 2.94459625e+02 -2.55986252e+02 -2.25541363e+01
        3.46560608e+02 -7.39733047e+01 -6.11878723e+02 5.47691589e+02
        -2.54440277e+02 3.89578430e+02 -1.97429924e+01 -3.02441120e+01
        -6.05380936e+01 1.19867584e+02 4.81274353e+02 -4.50405655e+01
        -1.40259216e+02 2.12232571e+01 1.59985626e+02 -2.17766678e+02
        2.56421692e+02 -2.65730560e+02 -4.84368706e+01 5.41689529e+01
        3.31454849e+01 4.55346985e+02 1.60499935e+01 -1.21101921e+02
        -3.88884552e+02 -4.94051880e+02 -2.23814362e+02 -1.55660095e+02
        -2.15649078e+02 -3.69188110e+02 -4.65110741e+01 1.38883423e+02
        1.71303986e+02 6.53557053e+01 -2.25296249e+02 -1.37362366e+02
        -1.08860826e+01 -4.89726028e+01 -1.87406326e+02 -1.33776230e+02
        -7.75556717e+01 -1.74358902e+02 7.00710907e+01 3.60966705e+02</_></support_vectors>
    <decision_functions>
      <_>
        <sv_count>52</sv_count>
        <rho>-6.8439932421071797e-02</rho>
        <alpha>
          5.0000000000000000e-01 1.0525960713336291e-01
          5.0000000000000000e-01 5.0000000000000000e-01
          5.0000000000000000e-01 5.0000000000000000e-01
          5.0000000000000000e-01 1.6492593552345239e-01
          5.0000000000000000e-01 5.0000000000000000e-01
          2.4772791491792806e-01 5.0000000000000000e-01
          1.1519019646476746e-02 3.8484042060328427e-01
          5.0000000000000000e-01 1.2934425740821587e-01
          4.9560752171466244e-01 5.0000000000000000e-01
          5.0000000000000000e-01 5.0000000000000000e-01
          5.0000000000000000e-01 5.0000000000000000e-01
          5.0000000000000000e-01 5.0000000000000000e-01
          5.0000000000000000e-01 5.0000000000000000e-01
          -5.0000000000000000e-01 -4.1553723332762957e-02
          -5.0000000000000000e-01 -5.0000000000000000e-01
          -3.3440622931480851e-01 -5.0000000000000000e-01
          -5.0000000000000000e-01 -3.0423595290159211e-01
          -5.0000000000000000e-01 -5.0000000000000000e-01
          -2.6063794466389045e-01 -5.0000000000000000e-01
          -5.0000000000000000e-01 -5.0000000000000000e-01
          -5.0000000000000000e-01 -5.0000000000000000e-01
          -5.0000000000000000e-01 -2.9092443842373899e-02
          -5.0000000000000000e-01 -5.0000000000000000e-01
          -5.0000000000000000e-01 -2.3137407739031729e-01
          -4.1040125530709265e-01 -4.2752305019454490e-01
          -5.0000000000000000e-01 -5.0000000000000000e-01</alpha>
        <index>
          0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24
          25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45
          46 47 48 49 50 51</index></_></decision_functions></my_svm>
</opencv_storage>
```

## A.2/ Deep Learning

### A.2.1/ Réseaux de neurones convolutifs (CNN)

**Couche de convolutions :**

L'objectif de la couche de convolutions est d'extraire des caractéristiques du volume d'entrée (une image de dimension "$W$x$H$" sur "$m$" canaux, une image couleur possédant "$m = 3$" canaux). Pour ce faire, un noyau de convolution de dimension "$D_K$x$D_K$" est appliqué en plusieurs point de l'image suivant une fenêtre glissante se déplaçant d'un pas "$s$" (*"stride"*). Cette convolution consiste tout d'abord en un produit scalaire entre un pixel de l'image et le facteur du noyau de convolution à l'emplacement correspondant, puis à sommer chacun de ces produits. Le résultat d'une convolution de la couche "$i$" à un emplacement donné de l'image, comme le montre la figure A.4, est donc un unique scalaire. L'ensemble des convolutions sur l'image représente une nouvelle image de dimension réduite "$W_i$x$H_i$".

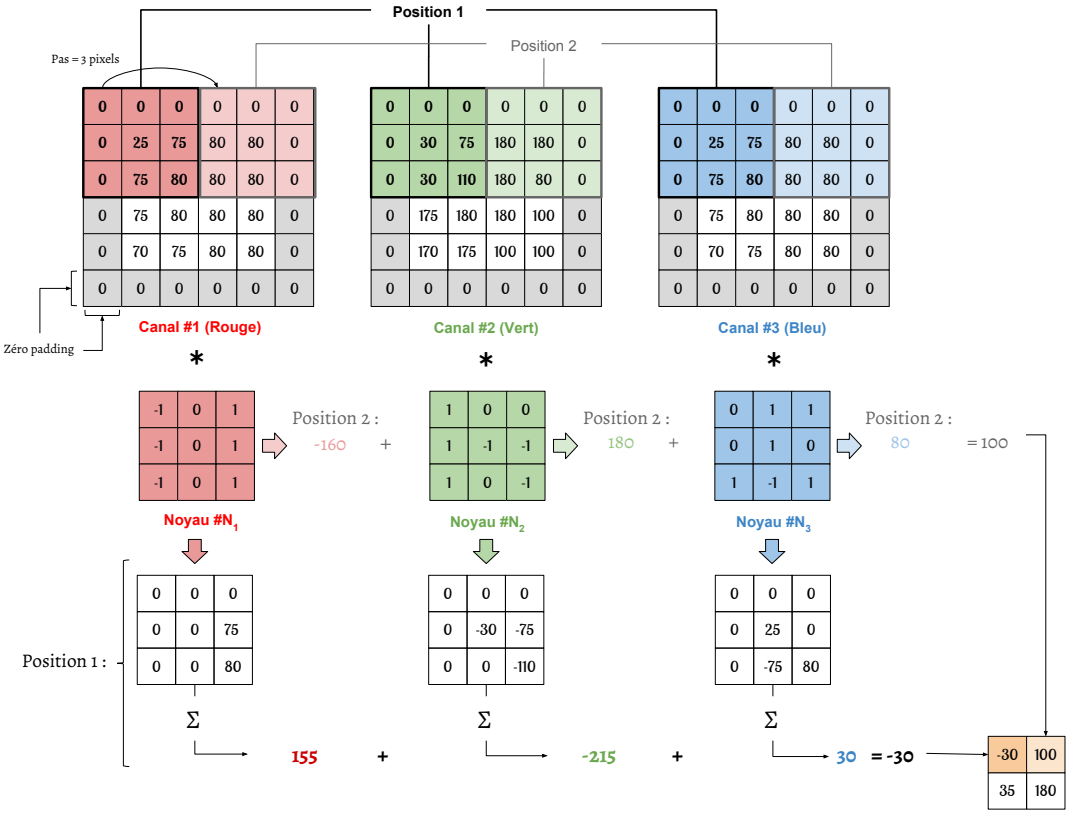



Figure A.4 – Illustration d'une convolution pour une image 4x4 de 3 canaux (RGB), un pas de 3 pixel, un *zero padding* et un noyau de convolution 3x3

Le pas, comme l'induit également la taille du noyau de convolution, a pour objectif de contrôler la dimension de l'image de sortie. Par exemple pour un pas de 2, le noyau de convolution se déplace de deux pixels entre deux positions de convolutions. Le pas est sélectionné de manière à produire une image aux dimensions entières (et non fractionnelles). Dans le cas d'un pas inférieur à la largeur du noyau de convolution, il y a alors un recouvrement avec des pixels utilisés à la position précédente du noyau. Afin de ne pas perdre l'information contenue sur les bords de l'image, les CNN utilisent le *"padding"*. Cette technique consiste à ajouter une ligne et une colonne à chaque bordure de l'image, permettant ainsi de considérer les pixels à la frontière de l'image. Ces nouveaux pixels sont généralement fixés à la valeurs zéro (*"zero padding"*).

L'image transférée en entrée du réseau comporte des pixels dont les valeurs sont comprises entre une borne inférieure et une borne supérieure (0 et 255 pour une image 8 bits). Hors, après une convolution, l'image résultante peut comporter des valeurs au delà de ces bornes. Afin de normaliser ces valeurs, on leur applique une fonction non linéaire d'activation pouvant prendre différentes formes. Comme le montre l'image A.5, les principales fonctions d'activations déterminent une nouvelle valeur en fonction de la valeur actuelle du pixel. La plus utilisée est la fonction "ReLU" (*"Rectified Linear Unit"*). Celle-ci est déterminée par la fonction "$max(0, x)$", transformant les valeurs négatives en zéros et conservant les autres valeurs. Cette fonction ne modifie pas la résolution spatiale des *feature maps* et ne comporte aucun paramètre du réseau.

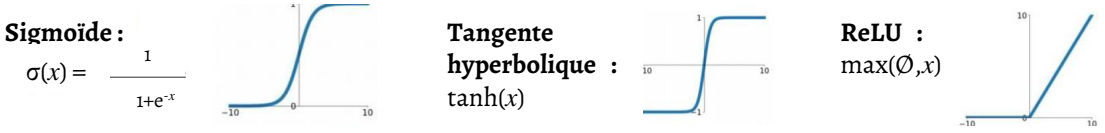



FIGURE A.5 – Exemples de fonctions non linéaires d'activations les plus connues

Les couches de convolution possèdent des hyperparamètres, c'est-à-dire des paramètres dont les valeurs sont fixées lors de la conception de l'architecture du réseau. Ils sont au nombre de quatre :

- le nombre de filtres "$K$",
- la taille $D_F$ des filtres (chaque filtre est de dimensions $D_F \times D_F \times m$ éléments),
- le pas "$s$" avec lequel on fait glisser la fenêtre correspondant au filtre sur l'image,
- le *zero-padding* "$P$".

Considérons une image (ou *feature map*), dans le réseau en entrée d'une couche de convolution. Cette image est de dimensions W×H×m, où "$W$" est sa largeur en pixels, "$H$" sa hauteur en pixels et "$m$" le nombre de canaux de l'image ($m = 1$ : image en niveaux de gris ; $m = 3$ : image couleur). Les paramètres, c'est à dire les poids et les biais qui sont ajustées lors de l'apprentissage, sont respectivement les éléments des noyaux de convolutions et le biais associé à chaque filtre. On dénombre ainsi "$D_F.D_F.K.m$" paramètres pour cette couche. Comme l'illustre la figure A.6, le résultat renvoyé par cette couche de convolution est une matrice de dimensions $W_C \text{x} H_C \text{x} D_C$, où :

$$W_C = \frac{W - D_F + 2.P}{s} + 1 \quad | \quad H_C = \frac{H - D_F + 2.P}{s} + 1 \quad | \quad D_C = m \tag{A.25}$$

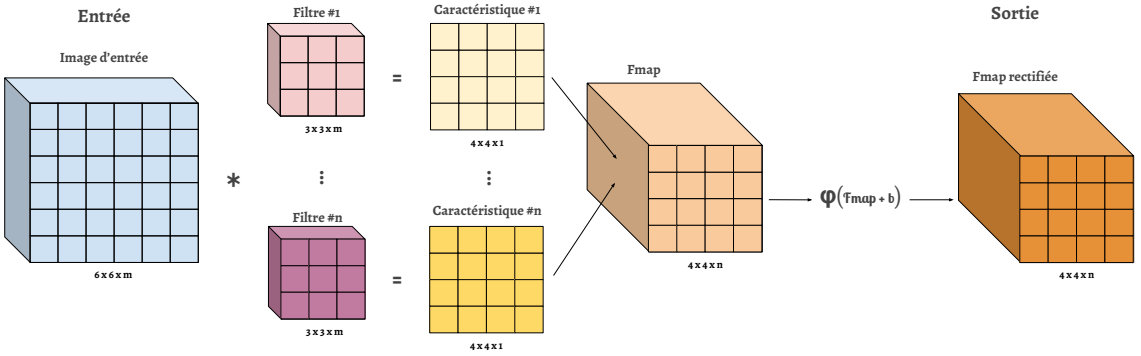



FIGURE A.6 – Illustration de l'ensemble convolution et activation

**Couche de sous échantillonnage *Pooling* :**
La partie de sous échantillonnage (ou *"pooling"*) a pour objectif de réduire la dimension spatiale d'une représentation d'entrée (image ou *feature map*), en termes de hauteur et de largeur (pas de profondeur), permettant de simplifier les caractéristiques contenues dans les sous-régions regroupées. La complexité de calcul du réseau diminue donc en réduisant le nombre de paramètres à apprendre dans les couches suivantes et le *"pooling"* produit une invariance en translation ainsi qu'une certaine maîtrise du sur-apprentissage. De la même manière que pour la convolution, une fenêtre glissante est déplacée sur l'image avec une certaine taille de pixels et un certain pas. Il existe différentes fonctions telles que le *"max pooling"* ou le *"average pooling"*, le plus utilisé étant le *"max pooling"*. Le *"max pooling"* sélectionne la valeur maximale présente dans la fenêtre, tandis que le *"average pooling"* calcul la moyenne des pixels dans la fenêtre. Cette fonction opère sur chaque canal de l'image indépendamment et ne requiert aucun paramètre.

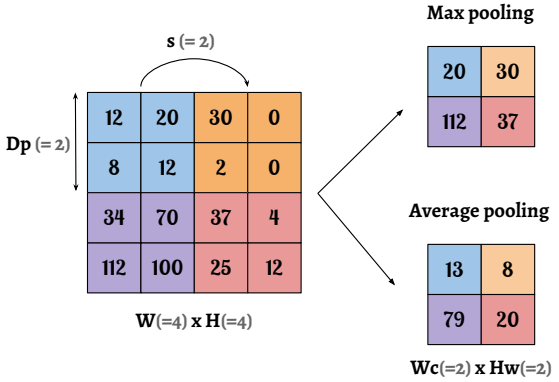



FIGURE A.7 – Illustration de l'étape de sous échantillonnage (*"pooling"*)

Tout comme la couche de convolutions, la couche de pooling possède des hyperparamètres qui sont au nombre de deux :

— la taille "$D_P$" des cellules : l'image est découpée en cellules carrées de taille $D_P \times D_P$ pixels,
— le pas "$s$" : les cellules sont séparées les unes des autres de S pixels

Pour chaque image ou *feature map* en sortie d'une couche de convolution, de taille W×H×m et passée à la couche de pooling, cette dernière renvoie une matrice de dimensions $W_P \text{x} H_P \text{x} D_P$, où :

$$W_C = \frac{W - D_P}{s} + 1 \quad | \quad H_P = \frac{H - D_P}{s} + 1 \quad | \quad D_P = m \tag{A.26}$$

**Couche de vectorisation *(Flattering)* :**
Après l'enchaînement d'un certain nombre de couches de convolutions/pooling, l'image d'entrée du réseau se retrouve être une *feature map* dont la largeur et la hauteur sont très petites par rapport à cette image d'entrée, et la profondeur (nombre de canaux) très grande. La classification s'effectue par une couche entièrement connectée, que nous verrons par la suite, qui ne prend pas en compte de structure spatiale. Il n'est donc plus nécessaire de conserver les informations sous la forme d'une *feature map*. Ainsi, une étape "d'aplatissement" ou de vectorisation est nécessaire afin de combiner les caractéristiques locales détectées par les couches précédentes. Comme le montre la figure A.8, les éléments de chaque canaux de la dernière *feature map* sont empilés pour créer un vecteur de données. Cette *feature map* en sortie de la couche "$n$" possède donc trois dimensions, $W_n \text{x} Hn \text{x} Dn$, et la phase de vectorisation la transforme en un vecteur de dimensions $W_n.Hn.Dn$.

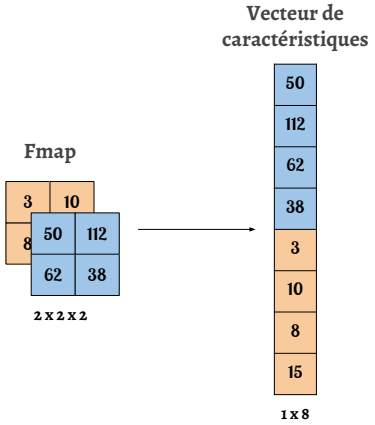



FIGURE A.8 – Illustration de l'étape de *"flattering"*

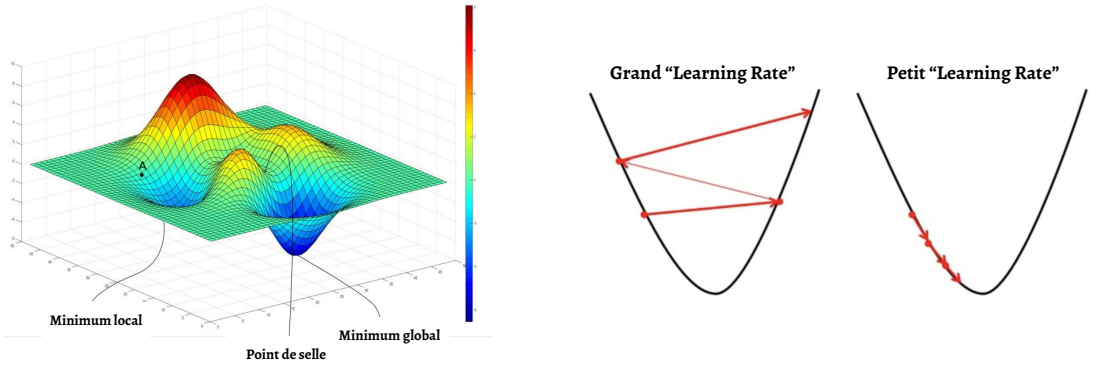
**Couche entièrement connectée (classification) :**
Comme leur nom l'indique, les couches entièrement connectées relient chaque neurone entre eux. Cet ensemble de couches entièrement connectées forment un réseau de neurones traditionnel. Pour effectuer une classification des caractéristiques générées par le réseau, qui ont alors été transformées en un vecteur de données, la dernière des couches utilise une fonction d'activation de type "*Softmax*", ou fonction de normalisation exponentielle. La fonction d'activation *Softmax* permet de normaliser les scores associés aux classes pour obtenir des valeurs comprises entre 0 et 1. La somme des résultats associés à chaque classe est égale à 1 et représente donc une probabilité associée à chacune de ces classes (distribution des probabilité). Cette fonction d'activation "$f_j(z)$" est présentée par l'équation A.27, où "$z$" est un vecteur de "$K$" éléments en sortie du réseau comportant les scores associés aux "$K$" classes, et $j$ correspond à la j$^{\text{ème}}$ classe du vecteur "$z$".

$$f_j(z) = \frac{e^{z_j}}{\sum_{k=1}^{K} e^{z_k}}, \quad \forall j \in [1;K] \text{ avec } \sum_{j=1}^{K} f_j(z) = 1 \tag{A.27}$$

**Optimisations de l'apprentissage :**
**Fonction de perte et descente de gradient *(gradient descent)* :**
Lors de l'apprentissage, en complément de la fonction finale d'activation *softmax*, une fonction de "perte", également appelée fonction de "coût", est utilisée afin de mesurer l'erreur d'apprentissage à une itération donnée et donc afin de quantifier précisément les performances du réseau à cet instant (écart entre probabilité de la prédiction et la vérité de terrain). L'objectif de la phase d'apprentissage est donc de trouver les paramètres optimums du réseau (les poids et les biais connectant chaque neurones) afin d'obtenir une classification optimale. Cette fonction de perte, le plus souvent étant soit l'erreur quadratique moyenne, soit l'entropie croisée (ou "*cross-entropy*"), est à minimiser au fur et à mesure des itérations d'apprentissage afin d'assister le réseau dans son apprentissage des caractéristiques. Ces fonctions sont présentées par l'équation A.28, où $MSE$ est l'erreur quadratique moyenne et $H(p,q)$ est l'entropie croisée, avec "$n$" le nombre de classes, "$q(i)$" est la probabilité de la prédiction de la classe "$i$" et "$p(i)$" est la vérité de terrain de cette même classe (représentation binaire de la classe "$i$" : 0 ou 1).

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(p(i) - q(x))^2 \qquad H(p,q) = -\sum_{i}^{n} p(i)log_2(q(i)) \tag{A.28}$$

Le réseau est initialisé avec un ensemble aléatoire de poids et de biais (décalage à une origine), conduisant à des performances peu pertinentes. Ces poids et ces biais sont alors ajustés avec la méthode de la descente de gradient. Afin de minimiser la fonction de perte, cet algorithme d'optimisation itératif ajuste les poids et les biais du réseau générant un déplacement vers un minimum local ou global du gradient de cette fonction d'erreur, tel que l'illustre la figure A.9a simplifiée, jusqu'à s'approcher au mieux de ce minimum (convergence). Le pas du déplacement à chaque itération est appelé le *"learning rate"*. Comme le montre la figure A.9b, s'il est grand, la méthode de la descente de gradient pourra couvrir une plus grande surface de la courbe sur la phase d'apprentissage, mais risque de manquer une pente vers un minimum, les variations pouvant être très brèves telle que le montre Li et al. [295]. Un petit learning rate est plus précis et permet de capter plus facilement les variations de pente mais est très chronophage du fait du coût calculatoire de cette optimisation qui dépend du nombre de paramètres du réseau. Un compromis est donc nécessaire.




(a) Illustration 3D de la descente de gradient (b) Effet du *Learning Rate*

FIGURE A.9 – Optimisation par descente de gradient

Il existe plusieurs types de descente de gradient, se différenciant principalement dans la quantité de données qu'ils utilisent (appelée *"batch"*). Les trois principaux étant les suivants :

- **le *batch gradient descent* :** l'erreur est calculée pour chaque images de l'ensemble d'apprentissage (*m* images) et le réseau n'est optimisé qu'après (batch=*m*),
- **la descente de gradient stochastique :** Les paramètres sont ajustés pour chaque image d'apprentissage (batch=1),
- **le *mini batch gradient descent* :** les données d'apprentissage sont divisées en groupes de *n* images, et l'optimisation intervient après le calcul de l'erreur de ces images (batch = n).

La taille du *mini batch* pour le *mini batch gradient descent* peut être ajustée en tant que paramètre d'apprentissage du réseau, au même titre que le *Learning Rate*. La quantité de fois où la totalité des images d'apprentissage sont présentées au réseau correspond au nombre d'étapes d'apprentissage (ou *"d'epoch"*), et le nombre d'itérations correspond au nombre de batch nécessaires pour compléter une epoch.

**Désactivation de neurones (*Dropout*) :**
Une couche entièrement connectée occupe la plupart des paramètres et les neurones développent par conséquent une interdépendance les uns avec les autres pendant l'entraînement, ce qui limite l'efficacité de chaque neurone. Cela conduit alors à un sur-apprentissage. Le "*dropout*" est une méthode de régularisation, permettant de forcer le réseau à "distribuer" la représentation des caractéristiques extraites de l'image au travers des différents neurones. Cette méthode "désactive" aléatoirement un nombre prédéfini de neurones lors de l'apprentissage. À chaque étape d'apprentissage, les neurones sont réactivés et de nouveaux sont sélectionnés aléatoirement pour être désactivés, forçant alors le réseau à apprendre une nouvelle représentation des données, maintenant la convergence même sans le dropout. Ce processus améliore la généralisation et préserve le réseau du sur-apprentissage. Lorsque le réseau est utilisé en tant que classifieur lors de la phase de test (appelée "inférence"), le dropout n'est nécessaire.

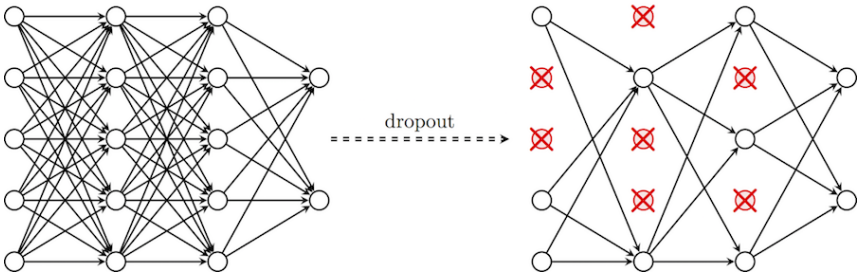



FIGURE A.10 – Illustration de l'optimisation par *"dropout"*

**Augmentation des données :**
L'avantage majeur du Deep Learning est la capacité des réseaux à accroître leurs performances proportionnellement à la quantité de données dédiées à l'apprentissage. Cependant, cette faculté est également un inconvénient, du fait qu'une grande quantité de données annotées est nécessaire, ce qui est parfois très difficile ou impossible à obtenir selon l'application. Dans le cas des CNN utilisant des images comme données d'entrées afin de généraliser une représentation des éléments d'une classe, il est possible d'augmenter grandement cette quantité de manière artificielle. Cette augmentation de données peut s'effectuer de différentes manières en appliquant certaines transformations aux images. On retrouve en effet de simples translations et rotations, des distorsions, des effet miroirs, l'extraction d'une certaine zone dans l'image, au même titre que des variations de contraste, de luminosité ou encore des variations de couleurs.

**Informations complémentaires :**
La figure A.11 illustre l'architecture de deux réseaux connus de la littérature : AlexNet [140], vainqueur du challenge ILSVRC (ImageNet challenge) de 2012 atteignant 16% d'erreur de classification avec $62,4.10^6$ paramètres, et VGGNet [229], vainqueur 2014 du même challenge avec moins de 10% d'erreur avec $138.10^6$ de paramètres.

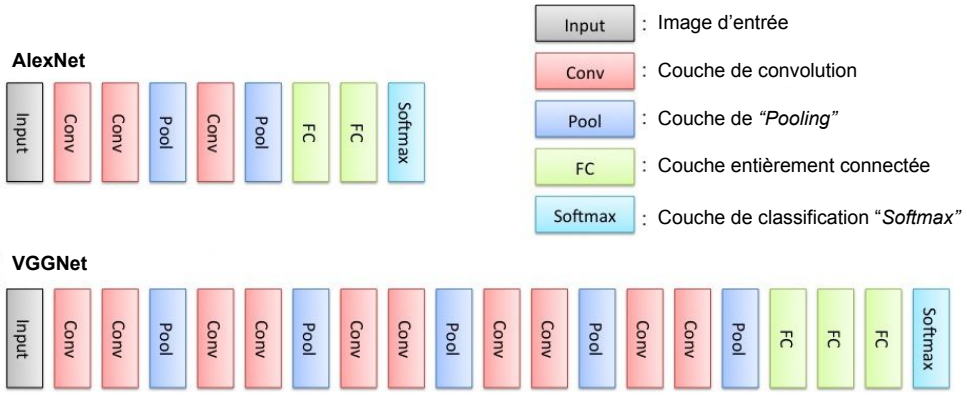
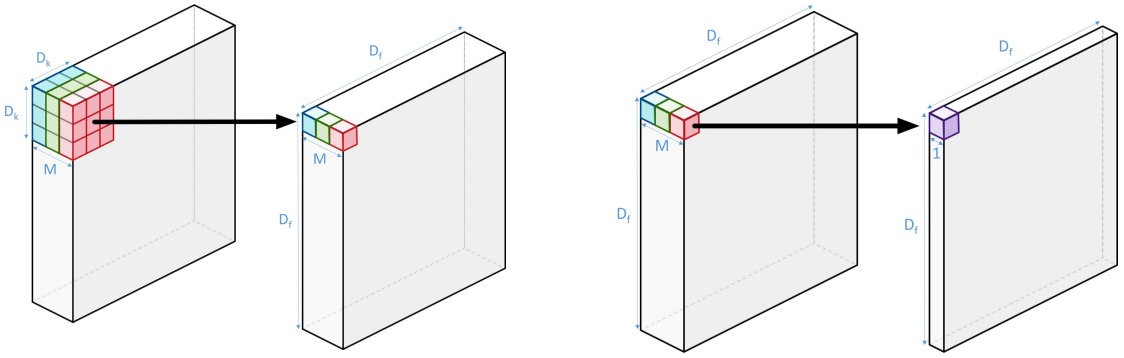



FIGURE A.11 – Architecture des réseaux "AlexNet" et "VGGNet" [293]

### A.2.2/ ARCHITECTURE MOBILENET

La particularité de MobileNet est de remplacer les convolutions conventionnelles par de nouvelles appelées *"Depthwise Separable Convolutions"*. Une convolution standard filtre les données d'entrées via une fenêtre glissante sur tous les canaux simultanément et combine les résultats en une nouvelle série de données. Avec un but identique, les convolutions *"depthwise separable"* divisent ces étapes en deux couches distinctes : une convolution appelée *"depthwise"* suivie d'une convolution 1x1 appelée *"pointwise"*. Cette factorisation permet de grandement réduire le nombre de calculs et la taille du modèle. Illustrée par la figure A.12a, la convolution *"depthwise"* applique un filtre sur chaque canal de l'image d'entrée. La convolution *"pointwise"*, figure A.12b, applique ensuite une simple convolution 1x1 afin de créer une combinaison linéaire de l'entrée.

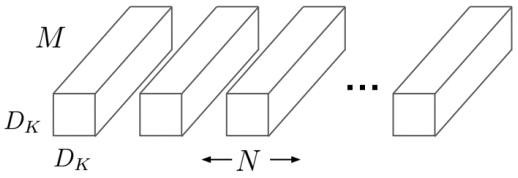



(a) Convolution "DepthWise"

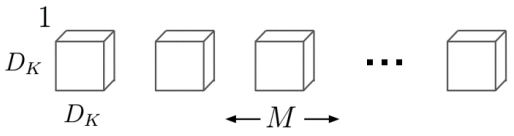
(b) Convolution "PointWise"

FIGURE A.12 – *"DepthWise Separable Convolution"* [2], $D_K = 3$, $M = 3$, $N = 1$

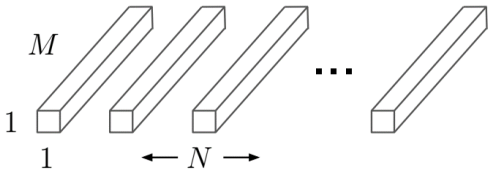
2. Image source : machinethink.net/blog/googles-mobile-net-architecture-on-iphone




(a) Principe d'une convolution standard




(b) Principe d'une convolution "depthwise"



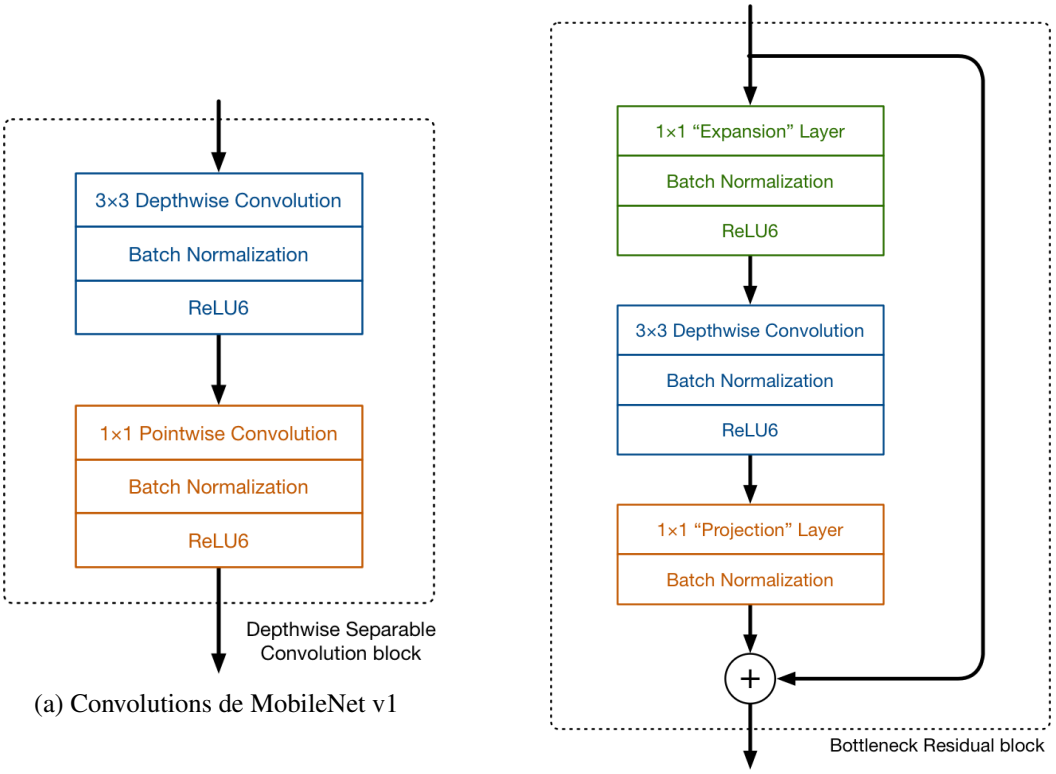

(c) Principe d'une convolution "PointWise"

FIGURE A.13 – Comparaison des principes des différentes convolutions [225]

Une comparaison directe entre les principes des convolutions classiques et des convolutions *"depthwise separable"* est illustrée figure A.13, où "$M$" correspond au nombre de canaux d'entrée (*"input depth"*, 3 pour une image couleur, dépend de la couche précédente pour une *"feature map"*), N au nombre de filtres (et donc nombre de canaux de sortie - *"ouput depth"*) et $D_K$ la taille du noyau de convolution. En considérant un pas de un pixel, le nombre de multiplications d'une convolution *"depthwise separable"* est définit par $[D_F.D_F.M.(N + D_K.D_K)]$, et $[D_F.D_F.M.N.D_K.D_K]$ définit le nombre de multiplications pour une convolution standard, où $D_F$ correspond à la taille de l'image traitée. On observe bien que les convolutions *"depthwise separable"* sont moins coûteuses en ressources. Les réseaux MobileNet utilisent des convolutions *"depthwise"* de taille $D_K = 3$.

Contrairement aux réseaux conventionnels, les réseaux MobileNet n'intègrent pas de couches de *"pooling"* entre ces blocs de convolutions. Ils utilisent également pour chacune de ces couches de convolutions une normalisation ainsi qu'une fonction d'activation de type ReLU (cf figures A.5 et A.6). La version 2 de MobileNet intègre quelques différences dans son bloc de convolution *"depthwise separable"*, illustrées figure A.14, et désormais appelé bloc de *"bottleneck"* résiduel *("residual bottleneck bloc")*. Le bloc de la version 2 contient une nouvelle première couche. Appelée couche d'expansion, c'est également une convolution 1x1 mais son objectif est d'étendre le nombre de canaux avant l'entrée de la convolution *"depthwise"*. Cette augmentation est gérée par un hyperparamètre appelé le facteur d'extension. Afin d'effectuer une somme entre l'entrée et la sortie de ce bloc via la connexion résiduelle, une dernière couche 1x1, appelée couche de projection, va réduire le nombre de canaux en sortie de la couche de convolution "depthwise", permettant de retrouver le même nombre de canaux qu'en entrée avant l'extension.

3. Image source : machinethink.net/blog/mobilenet-v2




(a) Convolutions de MobileNet v1

(b) Convolutions de MobileNet v2

FIGURE A.14 – Structure des convolutions des réseaux MobileNet v1 et v2 [3]

L'architecture du réseau MobileNet v1 est présentée par le tableau A.1a. En comptant les convolutions depthwise et pointwise comme des couches distinctes, ce réseau dispose de 28 couches cachées. En comparaison, l'architecture du réseau MobileNet v2, illustrée dans le tableau A.1b, est initialisée par une convolution complète, suivie par 19 couches de bottlenecks résiduels (remplacent du bloc de convolution depthwise separable de MobileNet v1). Dans ce tableau, le paramètre "$t$" représente le facteur d'extension de la première couche du bloc résiduel. Le paramètre "$c$" détermine le nombre de canaux de sortie de chaque séquence. Chaque couche présentée est répétée un certain nombre de fois déterminé par le paramètre "$n$". Le paramètre "$s$" détermine le "stride", c'est à dire le pas avec lequel le noyau de convolution se déplace sur l'image.

| Type/Stride | Filtre | Entrée |
|---|---|---|
| Conv / s2 | 3x3x3x32 | $224^2$x3 |
| Conv dw / s1 | 3x3x32 dw | $112^2$x32 |
| Conv / s1 | 1x1x32x64 | $112^2$x32 |
| Conv dw / s2 | 3x3x64 dw | $112^2$x64 |
| Conv / s1 | 1x1x64x128 | $56^2$x64 |
| Conv dw / s1 | 3x3x128 dw | $56^2$x128 |
| Conv / s1 | 1x1x128x128 | $56^2$x128 |
| Conv dw / s2 | 3x3x128 dw | $56^2$x128 |
| Conv / s1 | 1x1x128x256 | $28^2$x128 |
| Conv dw / s2 | 3x3x256 dw | $28^2$x256 |
| Conv / s1 | 1x1x256x256 | $28^2$x256 |
| Conv dw / s2 | 3x3x256 dw | $28^2$x256 |
| Conv / s1 | 1x1x256x512 | $14^2$x256 |
| 5x Conv dw / s1 | 3x3x512 dw | $14^2$x512 |
| 5x Conv / s1 | 1x1x512x512 | $14^2$x512 |
| Conv dw / s2 | 3x3x512 dw | $14^2$x512 |
| Conv / s1 | 1x1x512x1024 | $7^2$x512 |
| Conv dw / s2 | 3x3x1024 dw | $7^2$x1024 |
| Conv / s1 | 1x1x1024x1024 | $7^2$x1024 |
| Avg Pool / s1 | Pool 7x7 | $7^2$x1024 |
| FC / s1 | 1024x1000 | $1^2$x1024 |
| Softmax / s1 | Classifieur | $1^2$x1000 |

(a) Architecture des couches de MobileNet v1 [225]

| Type | Entrée | t | c | n | s |
|---|---|---|---|---|---|
| Conv2D | $224^2$x3 | - | 32 | 1 | 2 |
| Bottleneck | $112^2$x32 | 1 | 16 | 1 | 1 |
| Bottleneck | $112^2$x16 | 6 | 24 | 2 | 2 |
| Bottleneck | $56^2$x24 | 6 | 32 | 3 | 2 |
| Bottleneck | $28^2$x32 | 6 | 64 | 4 | 2 |
| Bottleneck | $14^2$x64 | 6 | 96 | 3 | 1 |
| Bottleneck | $14^2$x96 | 6 | 160 | 3 | 2 |
| Bottleneck | $7^2$x160 | 6 | 320 | 1 | 1 |
| conv2D 1x1 | $7^2$x320 | - | 1280 | 1 | 1 |
| avgpool 7x7 | $7^2$x1280 | - | - | 1 | - |
| conv2D 1x1 | 1x1x1280 | - | k | - | - |

(b) Architecture des couches de MobileNet v2 [226]

TABLE A.1 – Architecture des réseaux complets MobileNet v1 et v2 pour une image d'entrée de 224x224

# B

# Informations complémentaires

## Sommaire

## B.1/ Authentification de visages - Compression de données

### B.1.1/ Algorithmes étudiées

Afin de réduire les modèles de classification issus de RF, des SVM ou des CNN, nous avons évalué divers algorithmes de compression. Nous proposons ici de les présenter rapidement et d'exposer, table B.1, les taux de compression moyens de ces algorithmes sur ces modèles de classification.

- **Bzip2 :** L'algorithme bzip2 utilise la transformée de Burrows-Wheeler afin convertir toutes les séquences récurrentes en une chaîne de lettres identiques, puis les données résultantes sont une nouvelle fois transformées à l'aide du codage de Huffman.
- **Deflate :** Cet algorithme est une combinaison de l'algorithme de compression LZ77 et du codage de Huffman. Deflate64 est une variante de l'algorithme Deflate, utilisant entre autres un dictionnaire plus grand (64Ko au lieu de 32Ko).
- **LZMA :** Signifiant *"Lempel-Ziv-Markov chain Algorithm"*, c'est un algorithme de compression similaire à l'algorithme LZ77 avec une taille de dictionnaire variable (jusqu'à 4 Go). L'algorithme LZMA2 est un simple conteneur permettant d'inclure à la fois les données compressées par l'algorithme LZMA et les données non compressées.
- **PPMd :** Signifiant *"Prediction by Partial Matching by Dmitry"*, c'est un algorithme de compression basé sur l'algorithme PPM (*"Prediction by Partial Matching"*). Celui-ci est une technique adaptative de compression de données statistiques basée sur la modélisation et la prédiction de contexte. Les modèles PPM utilisent un ensemble de symboles parmi les symboles précédents non compressé pour prédire le symbole suivant.
- **RAR :** Signifiant *"Roshal ARchive"*, RAR est un format de fichier d'archive propriétaire prenant en charge la compression des données, la récupération des erreurs et l'étalement des fichiers. Structurellement, un fichier RAR est composé d'un marqueur ou d'un bloc d'introduction, d'un bloc d'archive qui comprend l'en-tête de l'archive et l'en-tête du fichier, et d'un bloc de fermeture contenant des commentaires supplémentaires ou d'autres informations nécessaires au traitement correct du fichier.

### B.1.2/ Comparaison des taux de compression

La table B.1 présente les taux de compression moyens de ces algorithmes appliqués aux modèles de classification RF et SVM avant la RPP, et aux CNN après leur quantification par la méthode *rounded weights*. D'une manière générale, ces algorithmes présentent des taux de compression proches pour chaque modèle. Cependant les algorithmes BZip2 et PPMd se démarquent pour les SVM/RF et nous avons sélectionné l'algorithme BZip2 pour nos expérimentations. Pour les CNN, c'est l'algorithme LZMA qui présente le meilleur taux de compression.

| Données | BZip2 | Deflate | Deflate64 | LZMA | PPMd | Rar |
|---|---|---|---|---|---|---|
| RF | 97,9 | 96,3 | 96,4 | 96,8 | 98,1 | 96,8 |
| SVM | 69,8 | 65,6 | 65,6 | 66,1 | 69,3 | 60,7 |
| MobileNet v1_025 | 74,7 | 70,8 | 71,4 | 76,3 | 75,7 | 72,2 |
| MobileNet v2_035 | 75,2 | 69,4 | 70,4 | 75,8 | 75,6 | 72,9 |

Table B.1 – Taux de compression moyens (en pourcent) d'algorithmes de courants pour la réduction de la taille de stockage des modèles de classification

## B.2/ Empreintes digitales et réseau vasculaire

### B.2.1/ Précisions sur les prétraitements

#### B.2.1.1/ Débruitage *(Non Local Means Denoising)*

Dans un cas très général inspiré par les filtres locaux par voisinage, on peut définir comme "voisinage d'un pixel $i$" tout ensemble de pixels $j$ de l'image, de sorte qu'une fenêtre autour de $j$ ressemble à une fenêtre autour de $i$. Tous les pixels dans ce voisinage peuvent être utilisés pour prédire la valeur de $i$. La figure B.1 illustre ce principe, où un fenêtrage sur un pixel (représentés respectivement par un rectangle et un rond bleus) représente le motif associé à ce pixel. Ce dernier est remplacé par la moyenne des fenêtres similaires (représentées par un encadré vert) dans son voisinage de recherche, correspondant à 5 fois son fenêtrage et centré sur le pixel en question.

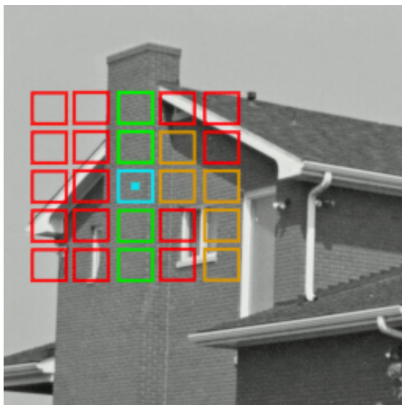


Figure B.1 – Débruitage par moyennes non locales - Recherche de similitudes par voisinage

Soient $v$ une zone observée dans l'image bruitée, définie dans un intervalle $\Omega \subset \mathbf{R}^2$, et $x \in \Omega$. L'algorithme des moyennes-NL estime la valeur de $x$ comme une moyenne des valeurs de tous les pixels dont le voisinage gaussien est similaire au voisinage de $x$, selon l'équation B.1. De plus amples informations concernant cet algorithme sont proposées en annexe B.2.1.1.

$$NL(v)(x) = \frac{1}{C(x)} \int_{\Omega} exp\left(-\frac{(G_a * |v(x+.) - v(y+.)|^2)(0)}{h^2}\right) v(y) d(y) \tag{B.1}$$

où :

- $G_a$ est un noyau gaussien dont la déviation standard est $a$
- $h$ agit comme un paramètre du filtrage (force du filtrage)
- $C(x) = \int_{\Omega} exp\left(-\frac{G_a * |v(x+.) - v(z+.)|^2)(0)}{h^2}\right) d(z)$ est le facteur de normalisation
- $(G_a * |v(x+.) - v(y+.)|^2)(0) = \int_{\mathbf{R}^2} G_a(t)\, |v(x+t) - v(y+t)|^2 d(t)$

#### B.2.1.2/ CLAHE *(Contrast Limited Adaptive Histogram Equalization)* :

L'égalisation adaptative d'histogramme (*Adaptive histogram equalization - AHE*), développée par Pizer et al. [296], est un algorithme d'amélioration de contraste adaptatif local. L'image est divisée en une grille de régions rectangulaires, et le nombre optimal de sous-régions dépend directement de l'image observée. Par conséquent, la détermination de ce nombre optimal de sous-régions requiert quelques expérimentations [297]. Pour chacune de ces régions, un histogramme est calculé et son égalisation est effectuée, basée sur les informations locales de l'image. De cet algorithme résulte une image pour laquelle le traitement de chaque pixel est différent et adapté à la distribution locale des intensités des pixels. Cette technique produit une image dans laquelle les différents objets, dont les valeurs des intensités se trouvent dans différentes plages d'intensités, se retrouvent visibles simultanément.
Cependant, l'amélioration du contraste est tellement importante que, dans des zones à faible signal (bruit de fond), la composante de bruit devient prépondérante [298]. De plus, pour certaines images, cette augmentation du contraste peut conduire à des discontinuités brutales des objets. Afin de remédier à ces difficultés, Pizer propose une variante à l'AHE, appelée "égalisation adaptatif d'histogramme avec limitation de contraste" (*Contrast-Limited Adaptive Histogram Equalization - CLAHE*). Avec cette méthode, l'amélioration de contraste produite localement est limitée par un paramètre (seuil) pouvant être ajusté suivant le type d'image. En d'autres termes, le nombre de pixels associés à chaque valeur d'intensité lumineuse dans l'histogramme local est tronqué à la valeur de ce seuil. Après cette troncature d'histogramme, les pixels dépassant ce seuil sont redistribués également au travers de l'histogramme, comme le montre la figure B.2.

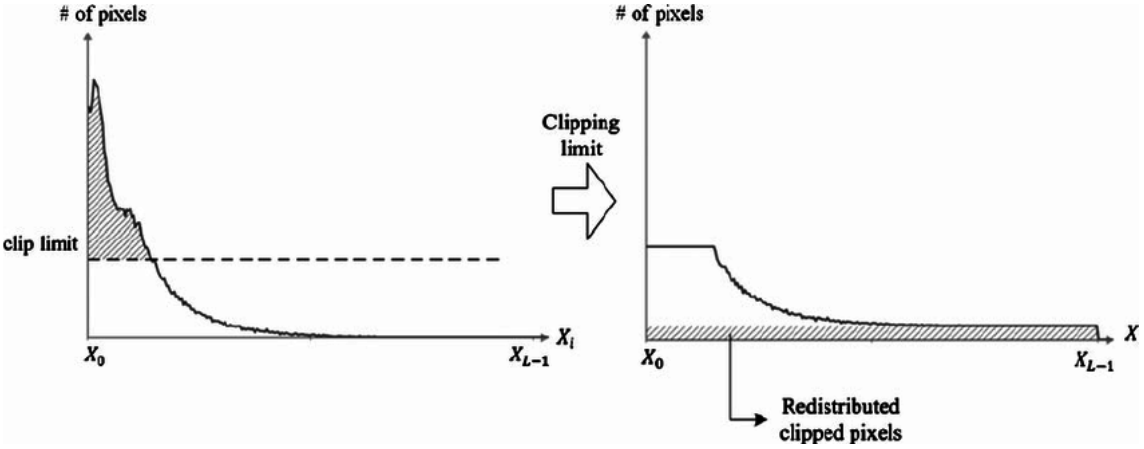



FIGURE B.2 – Clahe - Seuil de contraste et redistribution d'histogramme [294]

#### B.2.1.3/ INDICE DE POINCARE

De par la composition du motif des empreintes digitales, il est possible d'extraire des informations à différents niveaux de précision, comme il a été montré au début du chapitre 5. Le premier niveau consiste à détecter l'aspect général de l'empreinte. Différentes structures, appelées singularités, peuvent être observée et leur combinaison permet de distinguer des groupes d'empreintes digitales. Leur traitement peut s'effectuer à partir de la carte des orientations locales calculée lors du paramétrage du filtre de Gabor. Une nouvelle fenêtre glissante parcourt ces orientations. Cette fenêtre sous divisée en neuf régions (hauteur et largeur divisées par 3). Une matrice de taille 3 x 3 est alors crée en moyennant les orientations contenues dans chaque région de la fenêtre glissante. Ainsi, pour chaque position de la fenêtre glissante dans l'image, le type de singularité est estimée à l'aide de l'indice de Poincare [254] tel que décrit dans le paragraphe suivant [299].

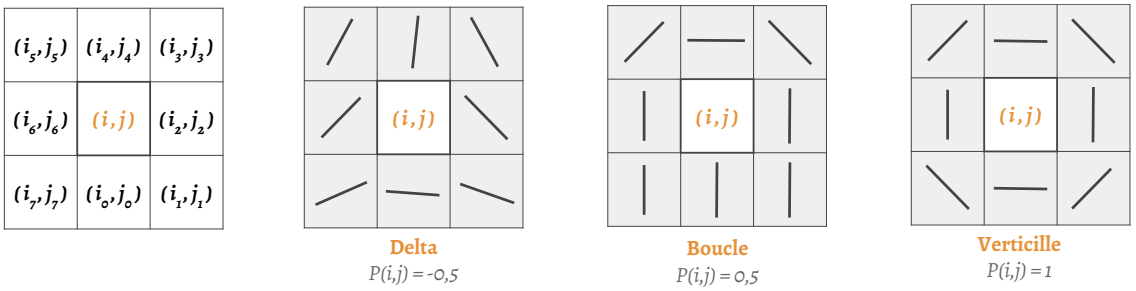



FIGURE B.3 – Exemple de singularités détectées par l'indice de Poincare selon les orientations

Soit $\theta(i, j)$ la valeur de l'orientation local moyennée à la position (i,j) correspondant à la position en laquelle est centrée la fenêtre glissante dans la carte des orientations locales. De par le procédé d'obtention de ces orientations, les valeurs sont comprises entre 0 et $2\pi$. Les 8 orientations voisines de l'élément central de la matrice sont prises en compte. Considérant la position $(i, j)$ comme étant l'élément central, et tel qu'illustré par la figure B.3, ces autres éléments sont définis par l'équation B.2. Pour chaque élément $k$, son orientation relative à l'orientation suivante est calculé. Ainsi, $\delta_k$ est définit par l'équation B.3.

$$\begin{array}{lll} * (i_0, j_0) = (i, j+1), & * (i_3, j_3) = (i+1, j-1) & * (i_6, j_6) = (i-1, j), \\ * (i_1, j_1) = (i+1, j+1) & * (i_4, j_4) = (i, j-1), & * (i_7, j_7) = (i-1, j+1). \\ * (i_2, j_2) = (i+1, j) & * (i_5, j_5) = (i-1, j-1), & \end{array} \tag{B.2}$$

$$\Rightarrow \begin{cases} \delta_k(i, j) = \theta(i_{k+1}, j_{k+1}) - \theta(i_k, j_k), & \forall k \in [0; 6] \\ \delta_7(i, j) = \theta(i_0, j_0) - \theta(i_7, j_7) \end{cases} \tag{B.3}$$

L'indice de Poincare, $P$, associé à l'élément $(i, j)$, est ensuite défini par :

$$P(i, j) = \frac{1}{2\pi} \sum_{k=0}^{7} \Delta_k(i, j) \,, \qquad \text{où} \qquad \Delta_k(i, j) = \begin{cases} \delta_k(i, j), & \text{si } |\delta_k(i, j)| < \frac{\pi}{2} \\ \pi + \delta_k(i, j), & \text{si } \delta_k(i, j) \leqslant -\frac{\pi}{2} \\ \pi - \delta_k(i, j), & \text{sinon} \end{cases} \tag{B.4}$$

| Verticille | Boucle | Delta | Types possibles |
|---|---|---|---|
| 1 | 0 | * | Verticille |
| 0 | 1 | * | Boucle, Arche |
| 0 | 2 | * | Double boucle ou Verticille |
| 0 | 0 | 0 | Arche |

TABLE B.2 – Table de référence des possible types d'empreintes digitales selon le nombre de singularités locales détectées de chaque type [254]. "*" signifie : aucune importance

La valeur de l'indice de Poincare définit une estimation sur le type de singularité présente à une certaine position. Quatre valeurs sont définies : $-0,5$ ; $0$ ; $0,5$ ; $1$. À une constante de tolérance prêt, la valeur $-0.5$ correspond à un "delta" local, la valeur $0.5$ représente une "boucle" locale, la valeur $1$ est associée à une "verticille" locale, et la valeur $0$ signifie qu'aucune singularité locale n'est présente. Les différentes singularités locales détectées permettent ensuite de définir l'aspect général de l'empreinte. Le décompte total de chaque singularité permet de faire correspondre l'empreinte à une certaine estimation présente dans la table de référence B.2. Une étape préliminaire de fusion des singularités voisines est nécessaire afin d'obtenir un décompte plausible.

#### B.2.1.4/ Seuillage adaptatif local

Il existe de nombreuses façons de fixer le seuil dans le cas d'un seuillage adaptatif local. Trois méthodes ont été présentées dans la section 5.2.2.2, et utilisées dans le chapitre 5. Ici, nous présentons des méthodes complémentaires à ces dernières.

— **Rappel de la définition du seuillage local :**

$$pixel(x,y) = \begin{cases} 255 & , \text{ si } \ pixel(x,y) \geq Seuil_{W_{(x,y)}} \\ 0 & , \text{ sinon} \end{cases} \tag{B.5}$$

— **Seuillage par médiane :**
Ici, "$\widetilde{W}_{(x,y)}$" représente la valeur médiane des pixels contenus dans la fenêtre centrée au pixel $(x,y)$, et "$c$" représente une constante d'ajustement.

$$\text{Seuil}_{W_{(x,y)}} \ = \ \widetilde{W}_{(x,y)} - \ c \tag{B.6}$$

— **Seuillage par moyenne d'extrémums :**
Le seuil est défini selon la moyenne des valeurs extrêmes de pixels (minimum et maximum) dans le voisinage du pixel $(x,y)$, et "$c$" représente une constante d'ajustement.

$$\text{Seuil}_{W_{(x,y)}} \ = \ \left[\frac{\min(W_{(x,y)}) + \max(W_{(x,y)})}{2}\right] - \ c \tag{B.7}$$

— **Seuillage de Bernsen :**
Bernsen [300] fixe un seuil local en fonction de la moyenne des extremums locaux. Le seuillage dépend ensuite d'une condition sur le contraste local. Celui-ci est comparé à un valeur définie par l'utilisateur (seuil de contraste). Si ce contraste est inférieur, le seuillage est défini par une comparaison entre le seuil local et une constante (luminance de valeur 128). Dans le cas contraire, ce seuil local est comparé à la valeur du pixel sur lequel est centrée la fenêtre de recherche.

$$\text{Seuil}_{W_{(x,y)}} \ = \ \left[\frac{\min(W_{(x,y)}) + \max(W_{(x,y)})}{2}\right]$$

**Si (Contraste local < Seuil contraste)**

$$pixel(x,y) = \begin{cases} 255 & , \text{ si } \ 128 \leqslant \text{Seuil}_{W_{(x,y)}} \\ 0 & , \text{ sinon} \end{cases} \tag{B.8}$$

**Sinon**

$$pixel(x,y) = \begin{cases} 255 & , \text{ si } \ pixel(x,y) > \text{Seuil}_{W_{(x,y)}} \\ 0 & , \text{ sinon} \end{cases}$$

— **Seuillage de Sauvola :**
La méthode de Sauvola [301] est une variation de la méthode de Niblack, conservant la structure principale du seuil. Le paramètre $r$ a été introduit, et correspond à la plage dynamique de la déviation standard. Le paramètre $k$ ne prend cette fois-ci que des valeurs positives.

$$\text{Seuil}_{W_{(x,y)}} = \overline{W}_{(x,y)}.\left[1 + k.\left(\frac{\sigma(W_{(x,y)})}{r} - 1\right)\right] \tag{B.9}$$

— **Seuillage de Phansalkar :**
La méthode de Phansalkar [302] est une variation de la méthode de Sauvola, dédiée à des images composées d'un très faible contraste. Une nouvelle composante est introduite, composée de deux paramètres $p$ et $q$, agissant sur l'exponentiel de la moyenne du fenêtrage local. Phansalkar recommande les valeurs suivantes : $k = 0.25$, $r = 0.5$, $p = 3$ et $q = 10$.

$$\text{Seuil}_{W_{(x,y)}} = \overline{W}_{(x,y)}.\left[1 + p.\mathrm{e}^{(-q.\overline{W}_{(x,y)})} + k.\left(\frac{\sigma(W_{(x,y)})}{r} - 1\right)\right] \tag{B.10}$$

### B.2.2/ Paramétrage des méthodes

#### B.2.2.1/ Optimisation des paramètres des algorithmes

Du fait des nombreuses étapes de traitement dans la chaîne algorithmique, notre étude est très paramétrique. Parmi les nombreuses combinaisons possibles, seul un certain nombre peut fournir des résultats optimaux. Par soucis de rapidité de développement, nous les avons fixés de manière expérimentale. Cependant ce paramétrage peut être optimisé de manière automatique afin d'augmenter la robustesse des prétraitements, comme décrit ci-dessous.

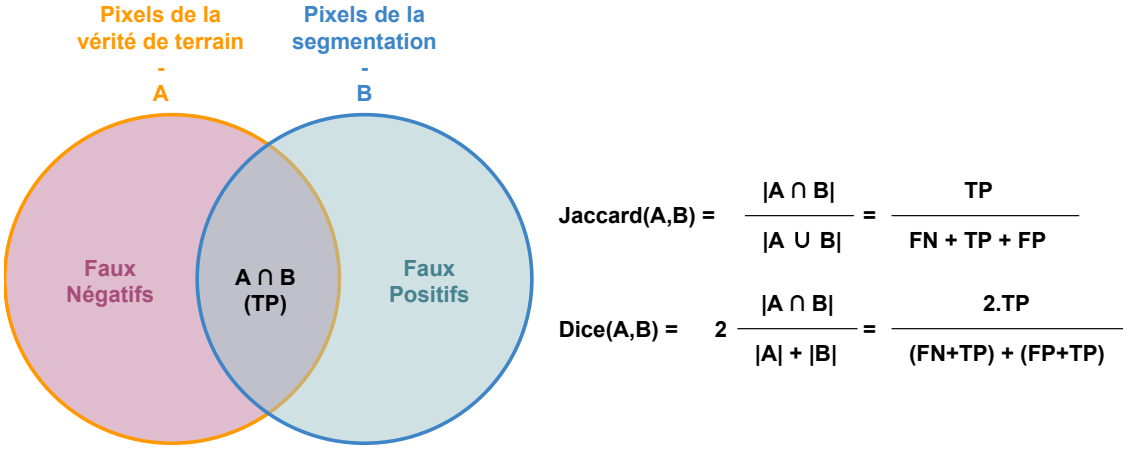



Figure B.4 – Estimation de la qualité de la segmentation automatique

Les performances et la fiabilité de la correspondance dépendent de la stabilité de la détection des points caractéristiques dans l'image, et cette détection dépend de la qualité de segmentation. Cette optimisation de paramètres nécessite la segmentation manuelle d'un certain nombre d'images (empreintes et veines), afin de constituer une base de données composée de la vérité de terrain de la segmentation et des images originales associées. Chacune de ces images originales est ensuite

segmentée automatiquement par notre algorithme, en faisant varier progressivement chaque paramètre. Les images résultantes sont ensuite comparées à leur vérité de terrain respective.
La qualité de la segmentation peut être évaluée en mesurant les coefficients "de Dice" [303] ou "de Jaccard", figure B.4. Pour chaque combinaison de paramètres, la mesure résultant de l'analyse des couples d'images est moyennée, et la plus haute valeur correspond au jeu de paramètres optimal, minimisant l'erreur de seuillage relative à la vérité de terrain.

#### B.2.2.2/ Paramètres utilisés dans cette étude

Comme précisé dans la section 5.2 du chapitre 5, la phase d'amélioration d'images est constituée d'une étape de correction d'images comprenant un débruitage, assurée par l'algorithme de débruitage par moyennes non locales, et d'une correction de contraste, effectuée par l'algorithme CLAHE. Ces traitements sont suivis d'un filtrage effectué par une banque de filtres de Gabor. Le paramétrage des différentes étapes est propre à chaque modalité. Le paramétrage de la correction d'image est résumé par la table B.3 et la table B.4 résume le paramétrage des filtres de Gabor.

| Traitement | Paramètre | Empreintes | Veines |
|---|---|---|---|
| **Débruitage** | **Taille du patch** | $5 \times 5$ | $3 \times 3$ |
| | **Fenêtre de recherche** | $15 \times 15$ | $15 \times 15$ |
| | **Intensité** | 5 | 10 |
| **CLAHE** | **Taille de la grille** | $8 \times 8$ | $8 \times 8$ |
| | **Contraste limite** | 3 | 4 |

Table B.3 – Paramétrage des traitements de la correction d'images

| Paramètre | Valeur |
|---|---|
| $W_\theta$ | 5 |
| $W_\phi$ | 5 |
| $W_F$ | 20 |
| $k$ | 0.65 |

(a) Empreintes digitales

| Paramètre | Valeur |
|---|---|
| Taille Noyau | $35 \times 35$ |
| $\sigma$ | 5 |
| $\lambda$ | 15 |
| $\gamma$ | 0.5 |
| $\psi$ | 0 |
| $K$ | 10 |

(b) Réseau vasculaire

Table B.4 – Paramétrage du filtrage de Gabor pour chaque modalité

Une seconde correction d'image suit l'étape de filtrage. Cette correction comporte également un débruitage, afin d'atténuer de possibles erreurs suite au filtrage, et une dernière correction de contraste par la méthode CLAHE. Le paramétrage de cette seconde correction d'images est défini par la table B.5. Pour la phase de segmentation, nous comparons deux méthodes de seuillage local et la méthode d'Otsu. Le tableau B.6 définit les paramètres fixés pour chaque méthode selon le type d'image seuillée. Le paramètre "surface minimale" définit le seuil au dessous duquel les zones segmentées sont supprimées.

| Traitement | Paramètre | Empreintes | Veines |
|---|---|---|---|
| **Débruitage** | **Taille du patch** | $5 \times 5$ | $5 \times 5$ |
| | **Fenêtre de recherche** | $20 \times 20$ | $20 \times 20$ |
| | **Intensité** | 20 | 20 |
| **CLAHE** | **Taille de la grille** | $8 \times 8$ | $9 \times 9$ |
| | **Contraste limite** | 2 | 4 |

TABLE B.5 – Paramétrage des traitements de la seconde correction d'images

| Type de seuillage | Paramètre | Empreintes | Veines |
|---|---|---|---|
| **Moyennes locales** | **Taille de la fenêtre** | $35 \times 35$ | $35 \times 35$ |
| | **Constante** | 2 | 4 |
| | **Surface min.** | 40 | 300 |
| **Niblack** | **Taille de la fenêtre** | $35 \times 35$ | $35 \times 35$ |
| | **K** | 0,3 | 0,2 |
| | **Constante** | 6 | 10 |
| | **Surface min.** | 135 | 300 |
| **Otsu** | **Surface min.** | 26 | 300 |

TABLE B.6 – Paramétrage des méthodes de seuillage comparées

#### B.2.2.3/ FILTRAGE DES VAISSEAUX SANGUINS

Nous proposons ici d'observer la décomposition de la phase de filtrage des vaisseaux sanguins par la banque de filtres de Gabor. Une série de 10 filtres, dont seule l'orientation varie, est appliquée à l'image corrigée après le débruitage et la correction de contraste. La figure B.5 illustre donc, pour chaque noyau, l'image résultante, puis, l'image finale résultant de la fusion de ces images intermédiaires suivant une règle de valeur minimale pour chaque pixel parmi ces images.

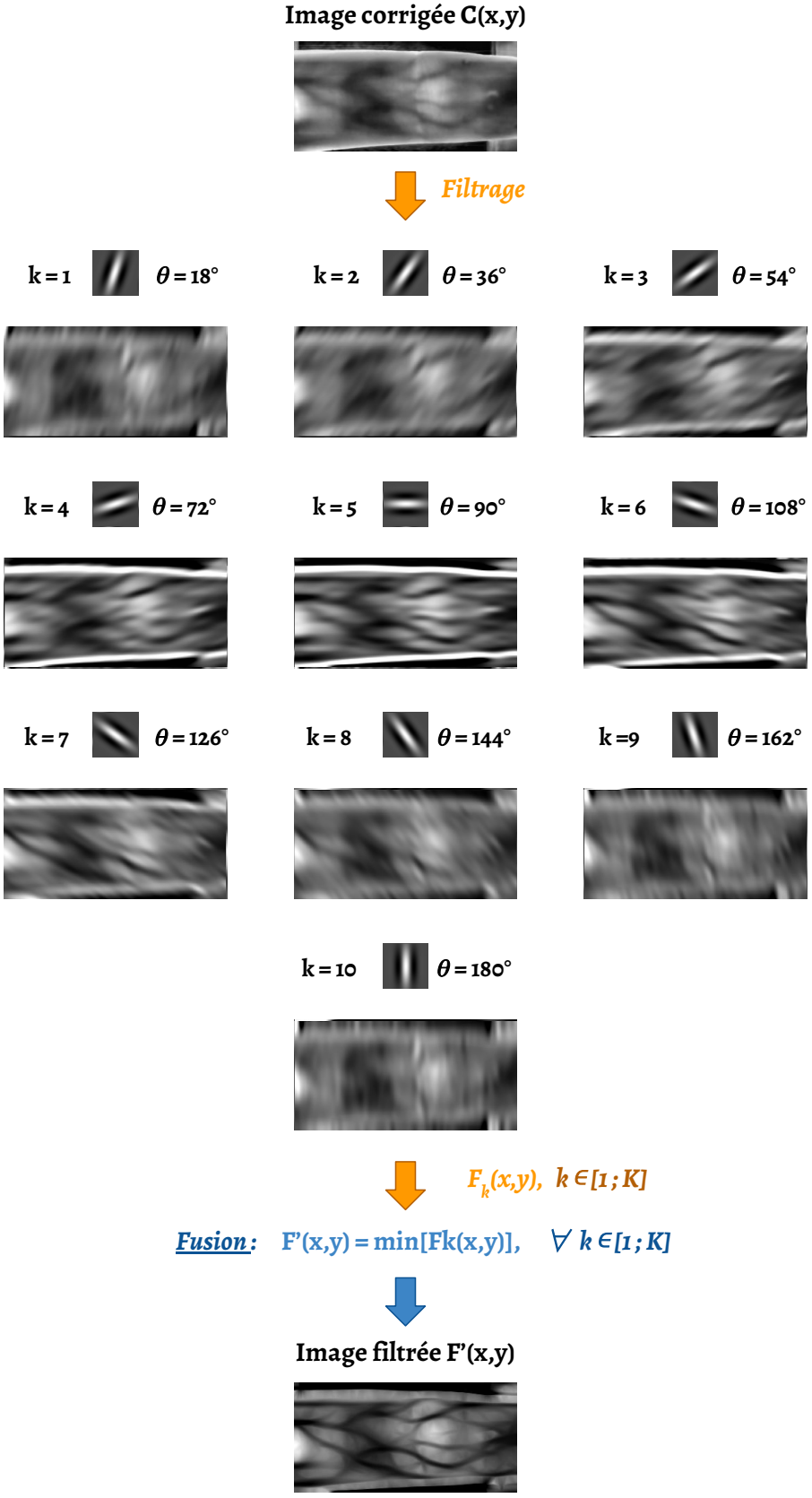



FIGURE B.5 – Décomposition des différents noyaux composant la banque de filtres de Gabor utilisée pour le filtrage du réseau vasculaire.

Document réalisé avec LaTeX et :
le style LaTeX pour Thèse de Doctorat créé par S. Galland — http://www.multiagent.fr/ThesisStyle
la collection de paquets `tex-upmethodology` — http://www.arakhne.org/tex-upmethodology/